



## **KERJA PRAKTIK – IF184801**

### **Rancang Bangun Aplikasi Toko Online Produk Kecantikan Berbasis Web**

Hexavara Technology

Jl. Ngaglik No. 50, Tambaksari, Kota Surabaya, Jawa Timur 60136

Periode: 1 Maret 2021 – 31 Mei 2021

#### **Oleh:**

Rosa Valentine Lammora 05111840000035

Afia Hana Yusriya 05111840000111

#### **Pembimbing Jurusan**

Ratih Nur Esti A., S.Kom., M.Sc.

#### **Pembimbing Lapangan**

Galang Amanda Dwi Pamungkas

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2021

*[Halaman ini sengaja dikosongkan]*



## **KERJA PRAKTIK – IF184801**

### **Rancang Bangun Aplikasi Toko Online Produk Kecantikan Berbasis Web**

Hexavara Technology

Jl. Ngaglik No. 50, Tambaksari, Kota Surabaya, Jawa Timur 60136

Periode: 1 Maret 2021 – 31 Mei 2021

#### **Oleh:**

Rosa Valentine Lammora 05111840000035

Afia Hana Yusriya 05111840000111

#### **Pembimbing Jurusan**

Ratih Nur Esti A., S.Kom., M.Sc.

#### **Pembimbing Lapangan**

Galang Amanda Dwi Pamungkas

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2021

*[Halaman ini sengaja dikosongkan]*

**LEMBAR PENGESAHAN  
KERJA PRAKTIK**

**Rancang Bangun Aplikasi Toko Online Produk  
Kecantikan Berbasis Web**

Oleh:

Rosa Valentine Lammora  
Afia Hana Yusriya

05111840000035  
05111840000111

Mengetahui,  
  
Pembimbing Lapangan  
  
Kerja Praktik

Menyetujui.  
  
Dosen Pembimbing  
  
Kerja Praktik



Galang Amanda Dwi Pamungkas    Ratih Nur Esti A., S.Kom., M.Sc.  
NIP. 198412102014042003

**SURABAYA**

**Juli, 2021**

*[Halaman ini sengaja dikosongkan]*

## **Rancang Bangun Aplikasi Toko Online Produk Kecantikan Berbasis Web**

Nama : Rosa Valentine Lammora  
NRP : 05111840000035  
Nama : Afia Hana Yusriya  
NRP : 05111840000111  
Departemen : Departemen Teknik Informatika,  
Fakultas Teknik Elektro dan  
Informatika Cerdas  
Pembimbing Departemen : Ratih Nur Esti A., S.Kom.,  
M.Sc.  
Pembimbing Lapangan : Galang Amanda Dwi  
Pamungkas

### **ABSTRAK**

*Sebuah bisnis memiliki kebutuhan untuk mempertahankan pelanggan melalui kanal-kanal promosi dan jual beli. Pelaku bisnis juga memerlukan sistem yang mempermudah manajemen usahanya agar memperoleh keuntungan dengan lebih efisien. Oleh karena itu, dibutuhkan suatu aplikasi yang dapat digunakan sebagai kanal jual beli dan alat manajemen usaha khusus untuk sebuah bisnis berbasis daring (e-commerce). Melalui kerja praktik ini, aplikasi tersebut dirancang untuk penggunaan berbasis web dengan kerangka kerja Laravel dan basis data MySQL. Dalam penggunaannya, pelanggan dapat melakukan pemesanan produk hingga transaksi. Sementara pelaku bisnis, melalui administratornya, dapat melakukan manajemen produk, pemesanan, transaksi, dan laporan sebagaimana dibutuhkan. Aplikasi ini diharapkan dapat secara efektif memenuhi kebutuhan usaha salah satu klien Hexavara Technology, yang merupakan toko produk kecantikan, untuk beroperasi secara daring.*

***Kata Kunci: e-commerce; Laravel; manajemen usaha; transaksi***

*[Halaman ini sengaja dikosongkan]*



## **KATA PENGANTAR**

Puji syukur kami haturkan kepada Tuhan Yang Maha Esa karena berkat rahmat-Nya kami dapat menyelesaikan salah satu kewajiban kami sebagai mahasiswa Departemen Teknik Informatika, yaitu Kerja Praktik yang berjudul: Rancang Bangun Aplikasi Toko Produk Kecantikan Online Berbasis Web. Kami menyadari kekurangan yang kami miliki baik dalam pelaksanaan kerja praktik maupun penyusunan buku ini, maka segala kritik, saran, dan komentar akan selalu kami terima. Namun, kami juga berharap bahwa buku laporan kerja praktik ini dapat menambah wawasan pembaca dan berguna sebagai sumber referensi.

Kami menyampaikan terima kasih kepada seluruh pihak yang telah membantu, memberikan saran dan bimbingan kepada kami dalam pelaksanaan kerja praktik hingga terselesaikannya buku ini, antara lain:

1. Orang tua penulis.
2. Bapak Ary Mazharuddin S., S.Kom., M.Comp.Sc. selaku koordinator kerja praktik.
3. Ibu Ratih Nur Esti A., S.Kom., M.Sc. selaku dosen pembimbing kerja praktik.
4. Bapak Galang Amanda Dwi Pamungkas dan rekan-rekan, selaku pembimbing lapangan kerja praktik kami di Hexavara Technology.

Surabaya, Juli 2021

Afia Hana Yusriya dan Rosa Valentine Lammora

*[Halaman ini sengaja dikosongkan]*

## DAFTAR ISI

LEMBAR PENGESAHAN KERJA PRAKTIK.....	5
ABSTRAK.....	7
KATA PENGANTAR.....	8
DAFTAR ISI.....	10
DAFTAR GAMBAR .....	14
DAFTAR TABEL.....	18
BAB I PENDAHULUAN .....	21
1.1    Latar Belakang.....	21
1.2    Tujuan .....	21
1.3    Manfaat.....	22
1.4    Rumusan Masalah .....	22
1.5    Lokasi dan Waktu Kerja Praktik.....	22
1.6    Metodologi Kerja Praktik .....	23
1.7    Sistematika Laporan .....	24
BAB II PROFIL PERUSAHAAN.....	27
2.1    Profil Hexavara Technology .....	27
2.2    Visi dan Misi Perusahaan .....	27
2.3    Lokasi.....	28
2.4    Struktur dan Organisasi dalam Lingkup Project .....	28
BAB III TINJAUAN PUSTAKA.....	29
3.1    HTML .....	29

3.2	CSS.....	29
3.3	Javascript.....	29
3.4	Laravel.....	30
3.5	MySQL.....	30
3.6	Visual Studio Code.....	30
3.7	Agile Scrum Framework.....	31
<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM .....</b>		<b>33</b>
4.1	Analisis Sistem .....	33
4.1.1	Deskripsi Umum Aplikasi.....	33
4.1.2	Analisis Kebutuhan.....	33
4.2	Perancangan Sistem.....	36
4.2.1	Diagram Kasus Penggunaan .....	36
4.2.2	Diagram Aktivitas.....	38
4.2.3	Conceptual Data Model .....	71
4.2.4	Physical Data Model.....	72
<b>BAB V IMPLEMENTASI SISTEM .....</b>		<b>73</b>
5.1	Implementasi Controller .....	73
5.1.1	Endpoint Customer .....	73
5.1.2	Endpoint Admin Penjual.....	79
5.2	Implementasi Antarmuka.....	82
5.2.1	Endpoint Customer .....	82
5.2.2	Endpoint Admin Penjual.....	91
<b>BAB VI PENGUJIAN DAN EVALUASI .....</b>		<b>97</b>

6.1	Tujuan Pengujian.....	97
6.2	Kriteria Pengujian.....	97
6.3	Skenario Pengujian .....	97
6.3.1	Pembeli .....	98
6.3.2	Penjual .....	98
6.4	Evaluasi Pengujian .....	99
BAB VII KESIMPULAN DAN SARAN .....		103
7.1	Kesimpulan.....	103
7.2	Saran.....	103
DAFTAR PUSTAKA.....		104
BIODATA PENULIS I .....		106
BIODATA PENULIS II .....		108

*[Halaman ini sengaja dikosongkan]*

## **DAFTAR GAMBAR**

Gambar 2.4.1 Struktur dan Organisasi dalam Lingkup Project...	28
Gambar 4.2.1 Diagram Kasus Penggunaan .....	37
Gambar 4.2.2 Diagram Aktivitas Registrasi Customer.....	38
Gambar 4.2.3 Diagram Aktivitas Login Customer .....	39
Gambar 4.2.4 Diagram Aktivitas Mencari Produk .....	40
Gambar 4.2.5 Diagram Aktivitas Melihat Daftar Produk Berdasarkan Kategori .....	40
Gambar 4.2.6 Diagram Aktivitas Melihat Detail Produk.....	41
Gambar 4.2.7 Diagram Aktivitas Menambah Produk ke Keranjang .....	42
Gambar 4.2.8 Diagram Aktivitas Melihat Keranjang .....	43
Gambar 4.2.9 Diagram Aktivitas Mengubah Kuantitas Produk di Keranjang .....	43
Gambar 4.2.10 Diagram Aktivitas Melakukan Checkout .....	45
Gambar 4.2.11 Diagram Aktivitas Menentukan Keterangan Pengiriman .....	46
Gambar 4.2.12 Diagram Aktivitas Menentukan Keterangan Pembayaran .....	47
Gambar 4.2.13 Diagram Aktivitas Melihat Riwayat Pemesanan	48
Gambar 4.2.14 Diagram Aktivitas Mengonfirmasi Pembayaran	49
Gambar 4.2.15 Diagram Aktivitas Memberi Penilaian dan Ulasan Produk Pesanan .....	50
Gambar 4.2.16 Diagram Aktivitas Melihat Detail Pemesanan ...	51
Gambar 4.2.17 Diagram Aktivitas Mengelola Profil .....	52
Gambar 4.2.18 Diagram Aktivitas Menambah Alamat Pengiriman .....	53
Gambar 4.2.19 Diagram Aktivitas Login Penjual.....	54
Gambar 4.2.20 Diagram Aktivitas Melihat Laporan .....	55
Gambar 4.2.21 Diagram Aktivitas Ekspor File Laporan .....	56
Gambar 4.2.22 Diagram Aktivitas Melihat Daftar Produk .....	57

Gambar 4.2.23 Diagram Aktivitas Menambah Produk.....	58
Gambar 4.2.24 Diagram Aktivitas Mengubah Data Produk .....	59
Gambar 4.2.25 Diagram Aktivitas Melihat Daftar Promo .....	60
Gambar 4.2.26 Diagram Aktivitas Menambah Promo.....	61
Gambar 4.2.27 Diagram Aktivitas Mengubah Promo .....	62
Gambar 4.2.28 Diagram Aktivitas Menghapus Promo .....	63
Gambar 4.2.29 Diagram Aktivitas Mem-publish Promo .....	64
Gambar 4.2.30 Diagram Aktivitas Melihat Daftar Produk di Landing Page.....	65
Gambar 4.2.31 Diagram Aktivitas Menambah Produk di Landing Page.....	66
Gambar 4.2.32 Diagram Aktivitas Mengubah Produk di Landing Page.....	67
Gambar 4.2.33 Diagram Aktivitas Menghapus Produk di Landing Page.....	68
Gambar 4.2.34 Diagram Aktivitas Melihat Daftar Order .....	69
Gambar 4.2.35 Diagram Aktivitas Mengubah Status Order .....	70
Gambar 4.2.36 Conceptual Data Model .....	71
Gambar 4.2.37 Physical Data Model .....	72
Gambar 5.2.1 Halaman Login .....	82
Gambar 5.2.2 Halaman Registrasi .....	82
Gambar 5.2.3 Halaman Awal .....	83
Gambar 5.2.4 Halaman Pencarian Kategori .....	84
Gambar 5.2.5 Halaman Detail Produk.....	85
Gambar 5.2.6 Halaman Keranjang .....	86
Gambar 5.2.7 Halaman Checkout 1: Keterangan Pengiriman ....	87
Gambar 5.2.8 Halaman Checkout 2: Pembayaran .....	87
Gambar 5.2.9 Halaman Checkout 3: Selesaikan Pembayaran ....	88
Gambar 5.2.10 Halaman Manajemen Profil .....	88
Gambar 5.2.11 Halaman Penilaian dan Ulasan.....	89
Gambar 5.2.12 Halaman Daftar Transaksi .....	90



Gambar 5.2.13 Halaman Overview Laporan Admin .....	91
Gambar 5.2.14 Halaman Manajemen Produk: Daftar Produk ....	92
Gambar 5.2.15 Halaman Manajemen Produk: Edit Produk .....	92
Gambar 5.2.16 Halaman Manajemen Landing Page: Daftar Produk yang Ditampilkan di Landing Page .....	93
Gambar 5.2.17 Halaman Manajemen Landing Page: Tambah Produk untuk Ditampilkan di Landing Page.....	93
Gambar 5.2.18 Halaman Manajemen Landing Page: Update Produk di Bagian Landing Page .....	94
Gambar 5.2.19 Halaman Manajemen Order: Daftar Order .....	94
Gambar 5.2.20 Halaman Manajemen Order: Keterangan Status Tracking Order .....	95
Gambar 5.2.21 Halaman Manajemen Promo: Daftar Promo .....	95
Gambar 5.2.22 Halaman Manajemen Promo: Tambah Promo ...	96

*[Halaman ini sengaja dikosongkan]*

## **DAFTAR TABEL**

Tabel 4.1.1 Kebutuhan Fungsional.....	35
Tabel 4.1.2 Kebutuhan Non-fungsional.....	36
Tabel 6.4.1 Evaluasi Pengujian .....	101

*[Halaman ini sengaja dikosongkan]*

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Sejak akhir tahun 2019, seluruh dunia telah digemparkan dengan semakin meningkatnya wabah Coronavirus Disease 2019 (COVID-19) tak terkecuali di Indonesia. Virus ini menyebabkan terhambatnya banyak aktivitas secara luring (luar jaringan) karena masyarakat harus mencegah persebaran virus. Oleh karena itu banyak aktivitas yang dialihkan menjadi daring (dalam jaringan) mulai dari kegiatan pembelajaran, seminar, dan juga jual beli.

Begitu pula yang terjadi pada salah satu klien Hexavara Technology yaitu pemilik suatu toko produk kecantikan. Klien Hexavara Technology tersebut awalnya adalah toko yang menjual kosmetik dan skincare secara luring. Akan tetapi, dengan adanya pandemi COVID-19, aktivitas toko secara luring menjadi terhambat. Sehingga pemilik toko selaku klien Hexavara Technology memutuskan untuk membuka toko secara daring. Untuk menyukseskan aktivitas toko tersebut, pemilik toko membutuhkan website khusus untuk melakukan aktivitas jual-belinya.

### **1.2 Tujuan**

Adapun tujuan dari pembuatan situs web jual beli ini adalah sebagai berikut:

- i. Menyelesaikan kewajiban kuliah Kerja Praktek di Institut Teknologi Sepuluh Nopember dengan bebas 2 SKS.
- ii. Untuk memberikan independensi pada salah satu klien Hexavara Technology dalam beroperasi sebagai toko online.

- iii. Untuk memenuhi kewajiban mata kuliah Kerja Praktek di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember.

### **1.3 Manfaat**

Manfaat yang dapat diperoleh dengan adanya situs web toko online antara lain:

- i. Salah satu klien Hexavara Technology selaku pemilik toko dapat melakukan kegiatan operasionalnya dengan penyesuaian yang dibutuhkan.
- ii. Meningkatkan *branding* toko online milik salah satu klien Hexavara Technology.
- iii. Memudahkan manajemen pesanan dan produk salah satu klien Hexavara Technology selaku pemilik toko.

### **1.4 Rumusan Masalah**

Berikut ini rumusan masalah pada Kerja Praktek pembuatan aplikasi toko online:

- i. Bagaimana solusi efektif dalam pembuatan aplikasi agar pembeli dapat melakukan pembelian dari toko dengan nyaman dan aman?
- ii. Bagaimana solusi efektif dalam pembuatan aplikasi agar penjual (pemilik toko) dapat memantau transaksi jual-beli di toko online-nya?

### **1.5 Lokasi dan Waktu Kerja Praktik**

Kerja praktik ini dilaksanakan pada waktu dan tempat sebagai berikut:

Lokasi	: Online
Waktu	: 1 Maret 2021 – 30 Mei 2021
Hari Kerja	: Senin – Jumat
Jam Kerja	: Fleksibel (minimal 12 jam per minggu)

## 1.6 Metodologi Kerja Praktik

Tahapan pengerjaan kerja praktik dapat dijabarkan sebagai berikut:

- i. Diskusi Kebutuhan Pengguna  
Tahap pertama yang dilakukan adalah memahami kebutuhan pengguna melalui diskusi dengan klien terkait permasalahan yang dimiliki dan solusi yang sudah terpikirkan. Sebagai bagian dari solusi tersebut, dijelaskan fitur-fitur yang diharapkan oleh klien. Dilakukan juga beberapa penyesuaian terhadap kebutuhan tersebut dengan batasan-batasan waktu dan biaya.
- ii. Eksplorasi *Tools*  
Selanjutnya dilakukan eksplorasi *tools* yang dapat digunakan untuk mengembangkan sistem. Dimulai dari desain rancangan database, desain antarmuka, penulisan kode, serta *plugin*, *extension*, atau aplikasi *third-party* yang dapat membantu pemenuhan kebutuhan pengguna.
- iii. Perancangan Sisten  
Perancangan sistem dilakukan berdasarkan kebutuhan pengguna yang telah didapatkan dan didiskusikan bersama klien. Maka dari itu, dibuatlah diagram yang meakili kebutuhan-kebutuhan tersebut yang telah disetujui. Di tahap ini, dibuat juga desain database dan antarmuka yang akan digunakan.

iv. Implementasi Sistem

Implementasi sistem didasarkan pada perancangan sistem yang sudah dibuat dan disetujui dengan menggunakan kerangka kerja Laravel. Pengerjaan sistem dilakukan secara iteratif sesuai metode Agile di mana setiap minggu terdapat beberapa target kebutuhan yang terpenuhi. Tim bekerja dan melaporkan perkembangannya setiap hari melalui *daily scrum meeting* (DSM). Satu siklus iterasi, yaitu satu minggu, diakhiri dengan *retrospective meeting*, di mana evaluasi keseluruhan pekerjaan dalam satu minggu dan penentuan target kebutuhan untuk siklus selanjutnya dilaksanakan.

v. Pengujian dan Evaluasi

Pengujian dilakukan dengan menguji fitur-fitur yang telah dibuat. Kesesuaian sistem dengan kebutuhan akan menentukan keberhasilan dalam pengujian. Hal ini akan menghasilkan hasil evaluasi, apakah sistem sudah sesuai dengan tujuan dan kebutuhan atau belum.

vi. Penyusunan Buku Laporan

Setelah masa kerja praktik selesai, tahap selanjutnya adalah penyusunan buku laporan kerja praktik sebagai dokumentasi sekaligus panduan pengembangan selanjutnya.

## 1.7 Sistematika Laporan

Laporan kerja praktik ini terdiri dari tujuh bab dengan rincian sebagai berikut:

i. Bab I Pendahuluan

Pada bab ini dijelaskan tentang latar belakang permasalahan, tujuan, waktu pelaksanaan, serta sistematika pengerjaan kerja praktek dan juga penulisan laporan kerja praktik.

ii. Bab II Profil Perusahaan



Pada bab ini dijelaskan secara rinci mengenai profil Hexavara Technology, tempat dilaksanakannya kerja praktek

iii. Bab III Tinjauan Pustaka

Pada bab ini dijelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian kerja praktek di Hexavara Technology

iv. Bab IV Analisis dan Perancangan Sistem

Pada bab ini dijelaskan gambaran umum sistem, proses bisnis sistem, kebutuhan sistem, serta desain sistem dari segi basis data.

v. Bab V Implementasi Sistem

Pada bab ini dijelaskan fitur-fitur yang dibuat saat proses kerja praktik serta hasil akhir pengerjaan kerja praktik.

vi. Bab VI Pengujian dan Evaluasi

Pada bab ini dijelaskan tentang hasil pengujian dan evaluasi sistem secara keseluruhan yang telah dibuat selama pelaksanaan kerja praktik.

vii. Bab VII Kesimpulan dan Saran

Pada bab ini dipaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan kerja praktik.

*[Halaman ini sengaja dikosongkan]*

## **BAB II**

### **PROFIL PERUSAHAAN**

#### **2.1 Profil Hexavara Technology**

Hexavara Technology merupakan perusahaan penyedia solusi teknologi informasi yang berkantor pusat di Surabaya dan resmi berdiri pada tahun 2017. Saat ini, Hexavara Technology telah melayani lebih dari 250 *project* dan aplikasi. Beberapa *project* antara lain adalah sistem informasi sekolah, sistem koperasi syariah, sistem administrasi manajemen portal truk, dan lain-lain.

#### **2.2 Visi dan Misi Perusahaan**

i. Visi

Menjadi perusahaan konsultan dan penyedia jasa pengembangan IT yang mengedepankan solusi yang terbaik dan komprehensif serta selalu melakukan riset guna menjaga kualitas yang baik dan berkesinambungan.

ii. Misi

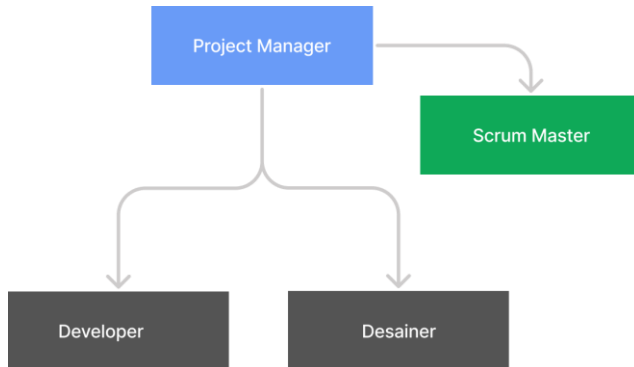
- a. Mengidentifikasi *problem*, menerjemahkan kebutuhan, serta memberikan solusi yang terbaik sesuai dengan kondisi *client* sebagai komitmen tinggi kami akan keterjagaan kualitas project/produk dan kepuasan *client*.
- b. Menjunjung tinggi etika serta kerjasama yang baik antar *stakeholder* terkait, masyarakat, maupun pemerintahan.
- c. Pengembangan perusahaan serta peningkatan mutu SDM yang berkesinambungan.

## 2.3 Lokasi

Hexavara Technology terletak di Jalan Ngaglik No.50, Tambaksari, Kota Surabaya, Jawa Timur 60136.

## 2.4 Struktur dan Organisasi dalam Lingkup Project

Pada perusahaan Hexavara Technology, mayoritas proyek dikerjakan menggunakan metode Agile Scrum yang dieksekusi oleh tim kecil. Tim ini terdiri dari setidaknya satu Project Manager, satu Scrum Master, beberapa Developer dan Desainer. Project Manager bertugas untuk menganalisis kebutuhan proyek dari *client*, merencanakan pengerjaan proyek, serta mengontrol waktu, pembiayaan, dan cakupan proyek. Scrum Master bertugas sebagai moderator pada *daily standup meetings*, di mana tim melakukan delegasi pekerjaan. Secara struktural, susunan tim diilustrasikan seperti pada Gambar 2.4.1.



Gambar 2.4.1 Struktur dan Organisasi dalam Lingkup Project

## **BAB III**

### **TINJAUAN PUSTAKA**

#### **3.1 HTML**

HyperText Markup Language (HTML) adalah bahasa markup standar untuk dokumen yang dirancang untuk ditampilkan di *web browser* [1]. Dalam penggunaannya, HTML dapat dibantu oleh teknologi seperti CSS dan JavaScript.

#### **3.2 CSS**

Cascading Style Sheet (CSS) adalah bahasa style sheet yang digunakan untuk menggambarkan presentasi dokumen yang ditulis dalam bahasa *markup* seperti HTML. Elemen-elemen yang terdapat pada HTML dapat dibuat bervariasi dengan gaya-gaya, animasi, ataupun transformasi yang ditulis dengan CSS [2]. CSS adalah teknologi landasan World Wide Web (WWW), bersama HTML dan Javascript.

#### **3.3 Javascript**

Javascript adalah bahasa pemrograman berorientasi obyek untuk pengembangan perangkat lunak, terutama pada halaman web. Namun, saat ini Javascript juga digunakan untuk pengembangan aplikasi *mobile*, desktop, dan sistem tertanam (*embedded systems*). Pada halaman web, Javascript digunakan untuk memperkaya interaktivitas antara komponen-komponennya dan pengguna dengan melakukan *scripting*. Masukan dan luaran dilakukan oleh sistem yang menaungi file Javascript [3], yaitu masukan dari *hardware* pengguna berupa *mouse* atau *keyboard* dan luaran di susunan HTML dan CSS pada halaman web.

### 3.4 Laravel

Laravel adalah kerangka kerja (*framework*) *open source* PHP gratis yang dibuat oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web mengikuti arsitektur *model-view-controller* (MVC). Laravel berada di bawah lisensi MIT dengan menggunakan GitHub sebagai tempat berbagi kode. Beberapa fitur Laravel antara lain berbagai cara untuk mengakses database relasional dan berbagai utilitas yang membantu dalam penerapan dan pemeliharaan aplikasi. Laravel menyediakan komponen-komponen untuk melakukan pekerjaan umum seperti autentikasi dan transaksi database dengan lebih mudah [4]. Laravel memiliki ekosistem yang sangat membantu pengembangan serta struktur yang cukup mudah dipahami sehingga pengerjaan dapat fokus ke *core business*.

### 3.5 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*) [5]. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya yaitu SQL. SQL (*Structured Query Language*) adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

### 3.6 Visual Studio Code

Visual Studio Code adalah perangkat lunak penyunting kode sumber yang dibuat oleh Microsoft untuk Linux, macOS, dan Windows. Visual Studio Code menyediakan fitur seperti *debugging*, *syntax highlighting*, dan *Git versioning*.

### **3.7 Agile Scrum Framework**

Agile adalah metode pengembangan perangkat lunak yang berpedoman pada pengembangan secara iteratif, di mana kebutuhan dan solusi dapat berevolusi seiring berjalannya organisasi. Metode ini mengutamakan kecepatan *delivery*, fleksibilitas terhadap perubahan, dan kolaborasi antara pengguna dan tim *cross-functional* yang mengembangkan perangkat lunak [6].

*[Halaman ini sengaja dikosongkan]*



## **BAB IV**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **4.1 Analisis Sistem**

Dalam mengembangkan aplikasi toko online ini, dilakukan analisis terhadap kebutuhan sistem yang selanjutnya dipenuhi melalui penentuan rancangan sistem. Bab ini akan membahas sistem secara umum, kebutuhan sistem, serta perancangan sistem dari kasus-kasus penggunaan hingga model data.

##### **4.1.1 Deskripsi Umum Aplikasi**

Aplikasi toko online ini adalah aplikasi *e-commerce* berbasis web. Sistem pada aplikasi ini dapat mengelola transaksi jual beli yang dilakukan antara toko dan pelanggannya dengan level kustomisasi tertentu yang memberikan independensi lebih bagi pemilik toko.

Kedua jenis pengguna, yaitu administrator dan pelanggan memiliki *endpoint* terpisah. Melalui administrator, sistem memungkinkan toko mengelola produk yang dijual dari level tertinggi, yaitu Kategori, hingga level terendah, yaitu Varian. Pelanggan dapat menempatkan pesanan produk yang kemudian akan diproses oleh administrator. Sejauh ini, pemrosesan yang dilakukan administrator mencakup konfirmasi bukti pembayaran hingga memasukkan informasi pelacakan pengiriman dalam bentuk URL. Selain itu, sistem dapat merumuskan laporan jual beli, baik dalam bentuk PDF maupun XLSX sesuai kebutuhan administrator.

##### **4.1.2 Analisis Kebutuhan**

Dalam sistem ini, terdapat kebutuhan-kebutuhan yang harus terpenuhi. Proses pengumpulan kebutuhan dilakukan oleh Project Manager yang mengumpulkan kebutuhan dari *client* dan

menyesuaikan dengan kapabilitas tim pengembang dan batasan waktu serta biaya. Seluruh kebutuhan dipetakan ke dalam dua jenis, yaitu kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional untuk sistem ini mencakup proses dan layanan yang dilakukan oleh sistem. Kebutuhan non-fungsional mencakup properti perilaku yang dimiliki oleh sistem.

#### 4.1.2.1 Kebutuhan Fungsional

Pada Tabel 4.1.1, terdapat deskripsi kebutuhan fungsional yang dipenuhi sistem:

Kode	Deskripsi Kebutuhan	Modul
F01	Melakukan autentikasi	Autentikasi
F02	Pembeli melakukan <i>two-factor account registration</i>	Autentikasi
F03	Penjual dapat mengelola kategori produk	Manajemen Produk
F04	Penjual dapat mengelola produk dan variannya	Manajemen Produk
F05	Penjual dapat mengelola brand	Manajemen Produk
F06	Penjual dapat mengelola landing page	Manajemen Produk
F07	Pembeli dapat melihat produk	Produk
F08	Pembeli dapat melakukan pencarian produk	Pencarian

F09	Pembeli dapat menambah produk ke keranjang	Transaksi
F10	Pembeli dapat melakukan <i>checkout</i> pesanan	Transaksi
F11	Pembeli dapat mengkonfirmasi pembayaran pesanan	Transaksi
F12	Pembeli dapat memberikan penilaian dan ulasan produk yang dibeli	Ulasan
F13	Pembeli dapat melihat riwayat pemesanan	Transaksi
F14	Penjual dapat mengubah status pengiriman dan pembayaran pesanan	Manajemen Transaksi
F15	Penjual dapat mengelola event promo	Manajemen Promo
F16	Penjual dapat melihat laporan transaksi dan produk	Laporan
F17	Pembeli dapat mengelola profil	Profil

*Tabel 4.1.1 Kebutuhan Fungsional*

#### **4.1.2.2 Kebutuhan Non-fungsional**

Pada Tabel 4.1.2, terdapat deskripsi kebutuhan non-fungsional yang dipenuhi sistem:

Kode	Deskripsi Kebutuhan	Kualitas
NF01	Hanya akun pengguna yang terautentikasi yang dapat melakukan transaksi	<i>Security</i>
NF02	Sistem dapat diakses 24 jam	<i>Availability</i>
NF03	Sistem dapat diakses pada web browser	<i>Portability</i>
NF04	Tampilan sistem responsif	<i>Portability</i>

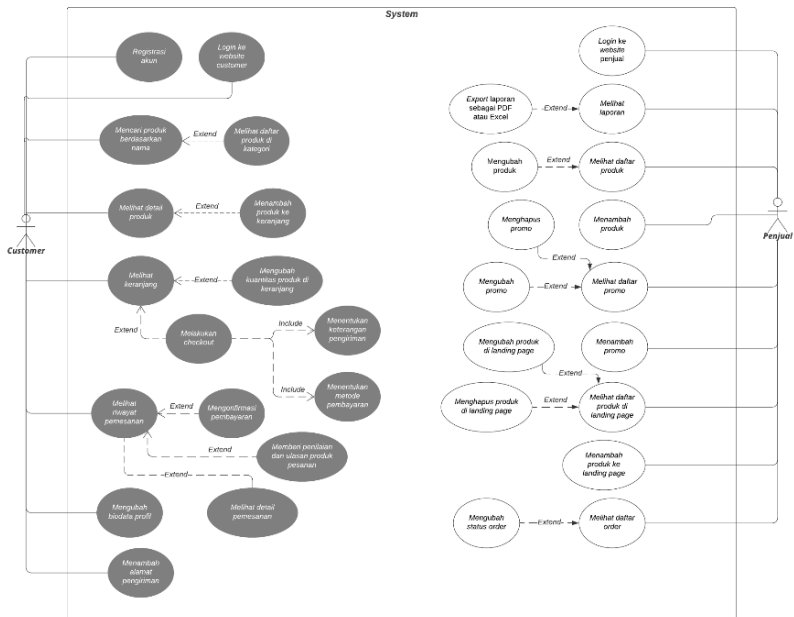
*Tabel 4.1.2 Kebutuhan Non-fungsional*

## 4.2 Perancangan Sistem

Setiap kebutuhan fungsional sistem dimodelkan menjadi kasus-kasus penggunaan. Agar proses yang dilewati pengguna dan sistem di tiap kasus penggunaan jelas, pada bab ini disediakan pula Diagram Aktivitas. Perancangan hubungan entitas data yang terdapat pada sistem dimodelkan melalui *Conceptual Data Model* dan *Physical Data Model*.

### 4.2.1 Diagram Kasus Penggunaan

Sistem memiliki kasus-kasus penggunaan seperti ilustrasi pada Gambar 4.2.1.

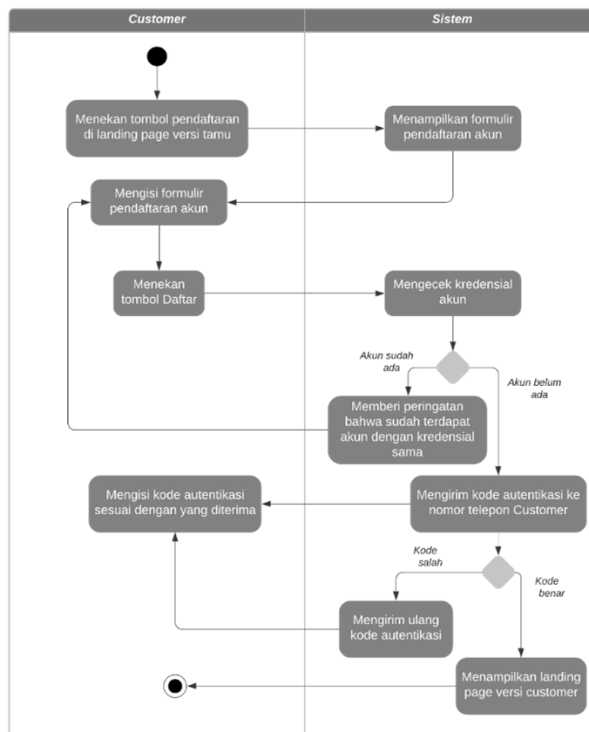


Gambar 4.2.1 Diagram Kasus Penggunaan

## 4.2.2 Diagram Aktivitas

### i. Registrasi Customer

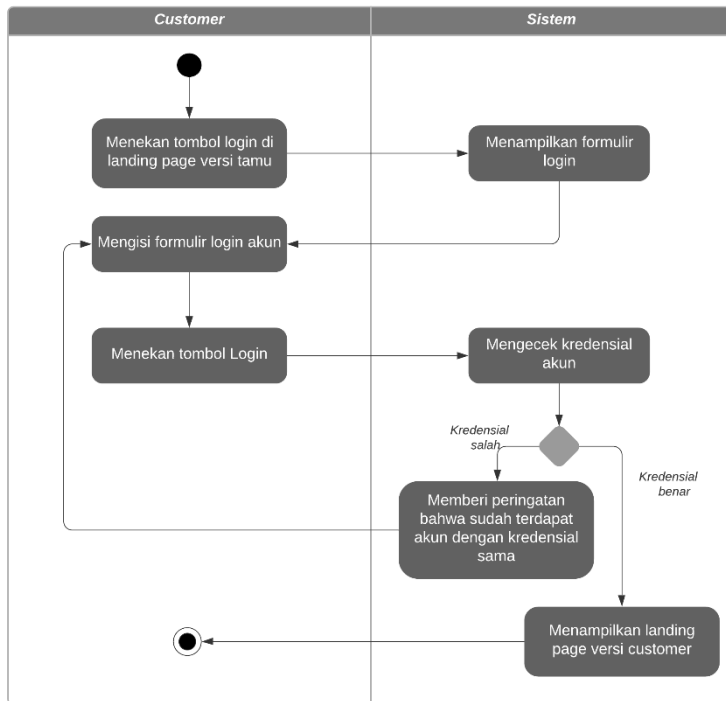
*Customer* dapat mendaftarkan akun baru. *Customer* akan diminta untuk memasukkan nomor telepon untuk keperluan verifikasi yang dilakukan saat itu juga melalui SMS. Verifikasi SMS ini dilakukan dengan memanggil fungsi API dari sistem eksternal, Nexmo, yang akan mengirimkan kode verifikasi ke nomor telepon customer. Kode verifikasi dihasilkan secara acak. Tahapan fitur registrasi customer dapat dilihat pada Gambar 4.2.2.



Gambar 4.2.2 Diagram Aktivitas Registrasi Customer

ii. Login Customer

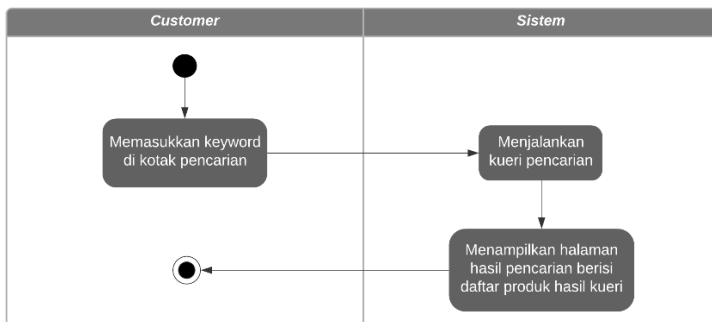
Setelah memiliki akun, *customer* dapat melakukan log in ke akun yang telah dibuat. Tahapan fitur login customer dapat dilihat pada Gambar 4.2.3.



Gambar 4.2.3 Diagram Aktivitas Login Customer

### iii. Mencari Produk

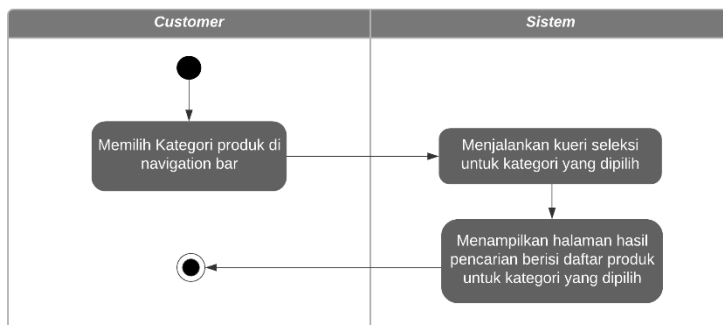
Fungsi pencarian pada aplikasi akan membaca keyword yang dimasukkan oleh *customer* dan menjalankan kueri seleksi dari database. *Customer* juga dapat melakukan *filtering* berdasarkan kategori dan merek. Tahapan fitur mencari produk dapat dilihat pada Gambar 4.2.4.



Gambar 4.2.4 Diagram Aktivitas Mencari Produk

### iv. Melihat Daftar Produk Berdasarkan Kategori

*Customer* dapat melakukan *filtering* pencarian produk berdasarkan kategori. Tahapan fitur melihat daftar produk berdasarkan kategori dapat dilihat pada Gambar 4.2.5.

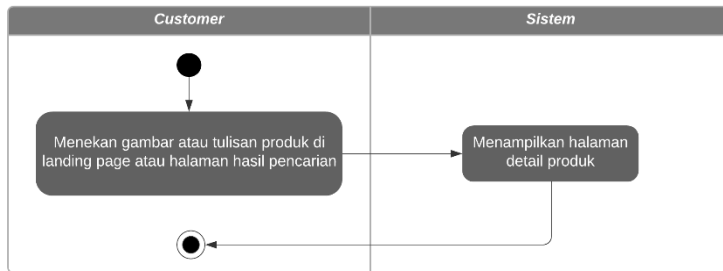


Gambar 4.2.5 Diagram Aktivitas Melihat Daftar Produk Berdasarkan Kategori



v. Melihat Detail Produk

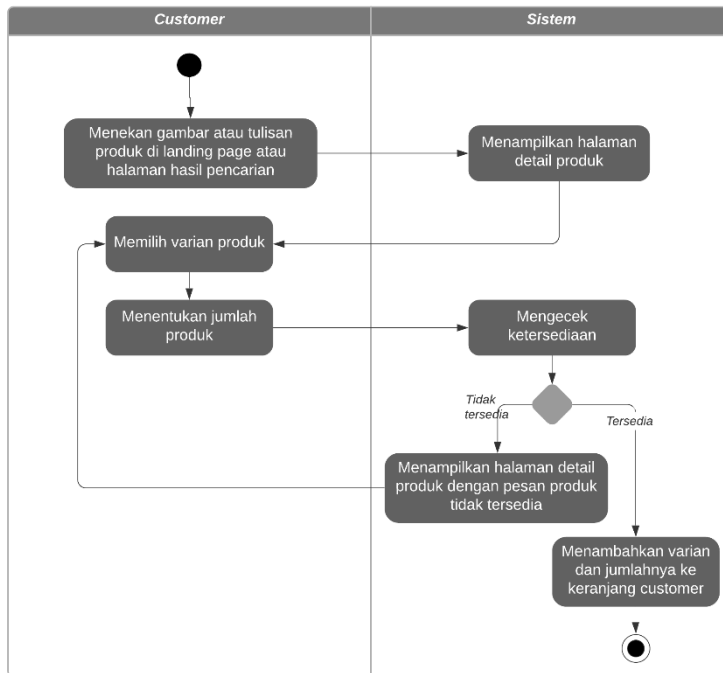
Detail produk akan memberikan informasi produk yang terdiri dari varian-varian produk, harga dan berat tiap varian, foto produk, dan pilihan untuk menambahkan varian terpilih ke keranjang *customer*. Tahapan fitur melihat detail produk dapat dilihat pada Gambar 4.2.6.



*Gambar 4.2.6 Diagram Aktivitas Melihat Detail Produk*

vi. Menambah Produk ke Keranjang

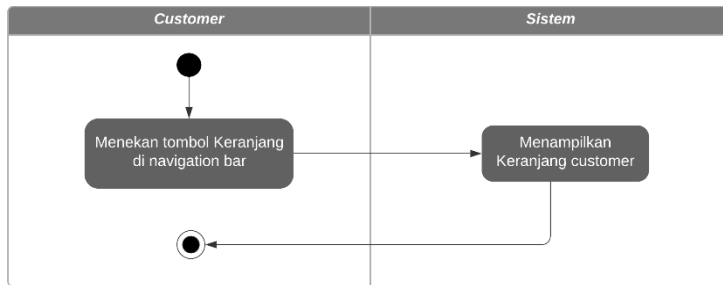
Keranjang customer menampilkan varian produk yang disimpan oleh *customer*. Selain melalui halaman detail produk, penambahan varian produk ke keranjang dapat dilakukan pula melalui halaman awal atau halaman hasil pencarian dengan menekan tombol keranjang di atas foto produk. Tahapan fitur menambah produk ke keranjang dapat dilihat pada Gambar 4.2.7.



Gambar 4.2.7 Diagram Aktivitas Menambah Produk ke Keranjang

vii. Melihat Keranjang

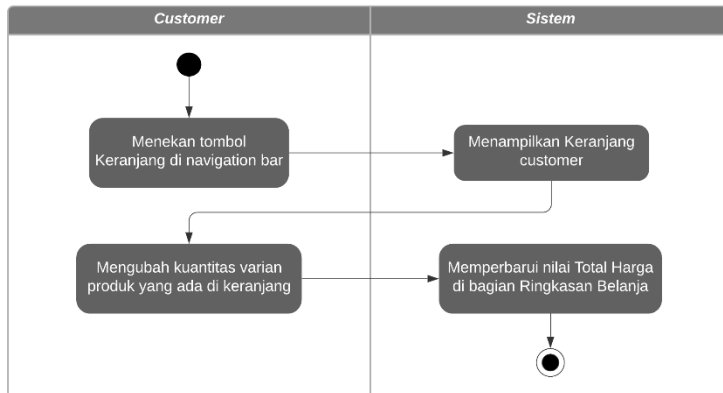
Selain melihat daftar produk yang disimpan di keranjang, *customer* juga dapat melihat perhitungan biaya pemesanan untuk produk-produk yang dipilih di halaman keranjang. Tahapan fitur melihat keranjang dapat dilihat pada Gambar 4.2.8.



Gambar 4.2.8 Diagram Aktivitas Melihat Keranjang

viii. Mengubah Kuantitas Produk di Keranjang

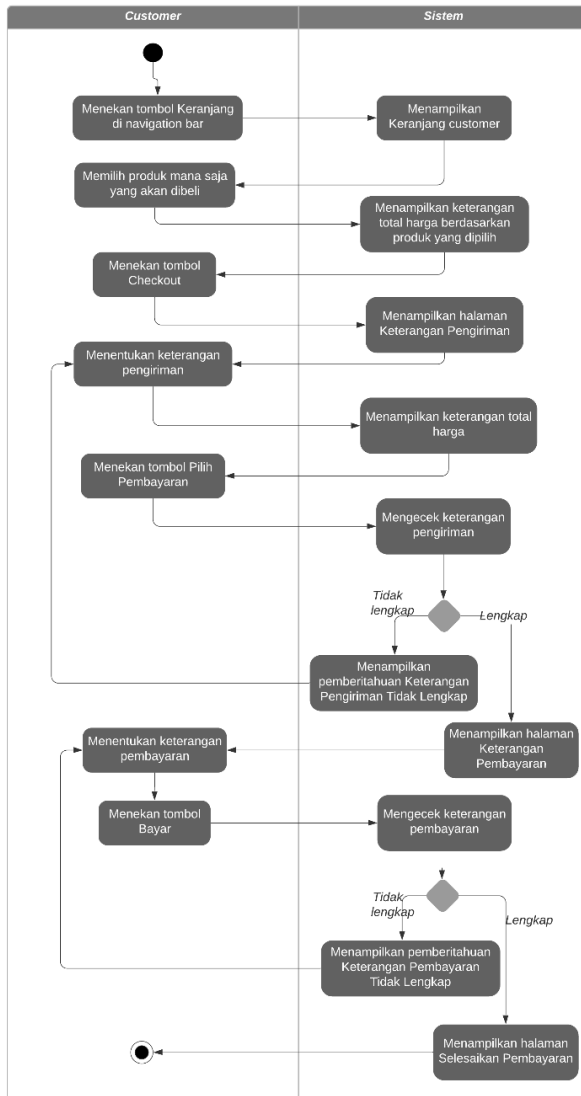
Kuantitas tiap varian yang terdapat di keranjang dapat diubah atau dihapus dari keranjang. Tahapan fitur mengubah kuantitas produk di keranjang dapat dilihat pada Gambar 4.2.9.



Gambar 4.2.9 Diagram Aktivitas Mengubah Kuantitas Produk di Keranjang

ix. Melakukan Checkout

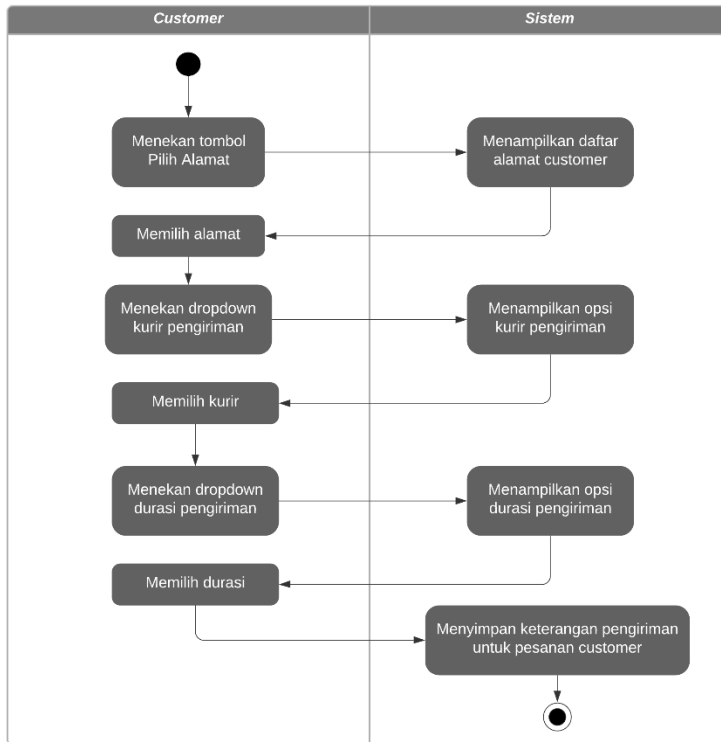
*Checkout* varian produk terpilih di keranjang dilakukan dalam dua tahap, yaitu pengumpulan informasi pengiriman kemudian informasi pembayaran oleh *customer*. Stok produk akan berkurang untuk sementara saat *checkout* berlangsung hingga selesai. *Customer* akan diberikan waktu untuk menyelesaikan pembayaran. Setelah pembayaran dilakukan di luar sistem, *customer* diarahkan untuk mengonfirmasi pembayaran dengan mengunggah bukti pembayaran ke sistem. Status pesanan dan pembayaran akan berubah seiring pembaruan dari *customer*. Tahapan fitur melakukan checkout dapat dilihat pada Gambar 4.2.10.



Gambar 4.2.10 Diagram Aktivitas Melakukan Checkout

x. Menentukan Keterangan Pengiriman

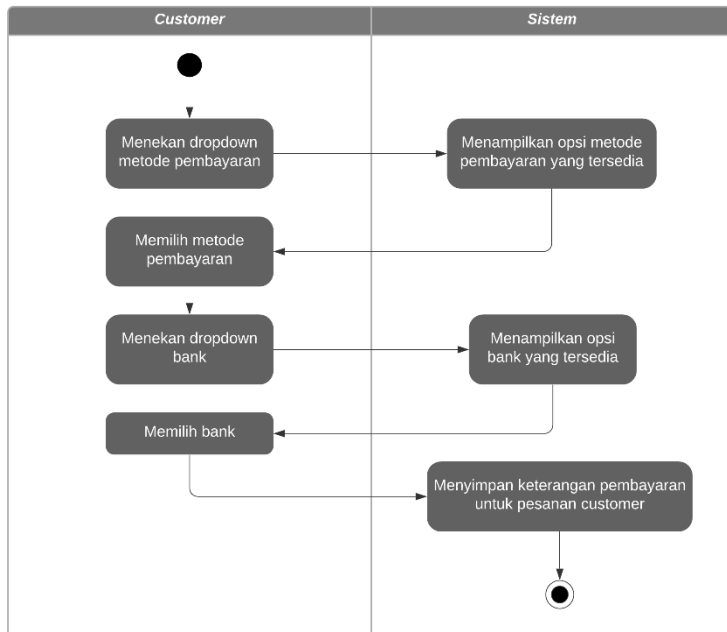
Saat melakukan *checkout*, *customer* menentukan alamat tujuan pengiriman pesanan dan kurir yang dipilihnya. Tahapan fitur menentukan keterangan pengiriman dapat dilihat pada Gambar 4.2.11.



Gambar 4.2.11 Diagram Aktivitas Menentukan Keterangan Pengiriman

xi. Menentukan Keterangan Pembayaran

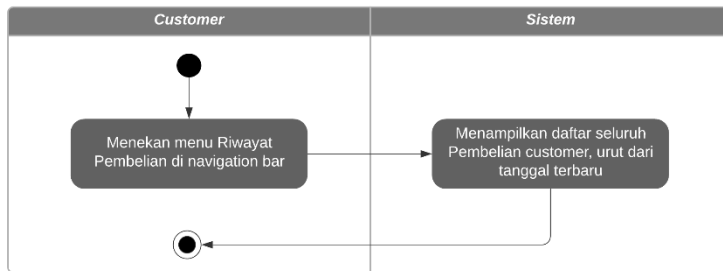
Saat melakukan *checkout*, setelah menentukan keterangan pengiriman, *customer* juga memberikan detail pembayaran yang dipilihnya. Tahapan fitur menentukan keterangan pembayaran dapat dilihat pada Gambar 4.2.12.



Gambar 4.2.12 Diagram Aktivitas Menentukan Keterangan Pembayaran

xii. Melihat Riwayat Pemesanan

*Customer* dapat melihat riwayat transaksinya. Secara *default*, riwayat yang ditampilkan adalah seluruh pesanan, baik yang sudah selesai maupun yang masih diproses. *Customer* juga dapat mengimplementasikan *filtering* berdasarkan tanggal pemesanan. Tahapan fitur melihat riwayat pemesanan dapat dilihat pada Gambar 4.2.13.

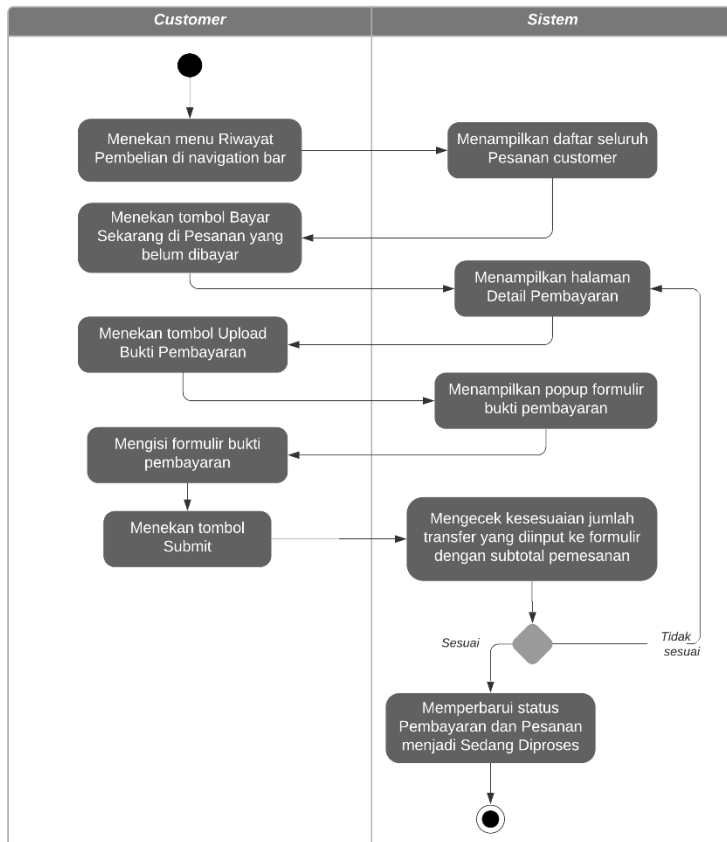


Gambar 4.2.13 Diagram Aktivitas Melihat Riwayat Pemesanan



xiii. Mengonfirmasi Pembayaran

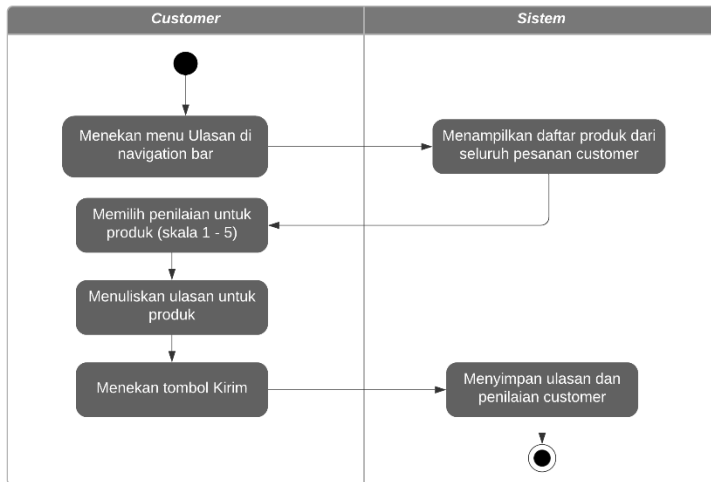
*Customer* melakukan konfirmasi pembayaran dengan mengunggah bukti pembayaran dan mengisi formulir keterangan detail pembayaran. Tahapan fitur mengonfirmasi pembayaran dapat dilihat pada Gambar 4.2.14.



Gambar 4.2.14 Diagram Aktivitas Mengonfirmasi Pembayaran

xiv. Memberi Penilaian dan Ulasan Produk Pesanan

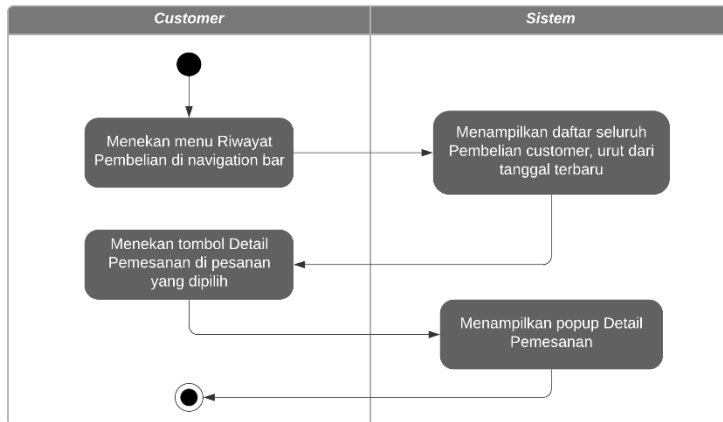
Setelah pesanan selesai dan produk diterima oleh *customer*, ia dapat memberikan penilaian dan ulasan terhadap produk yang telah diterima pesannya. Setiap penilaian yang diterima produk akan dikalkulasikan dan ditampilkan nilai rata-ratanya di sistem. Tahapan fitur memberi penilaian dan ulasan produk pesanan dapat dilihat pada Gambar 4.2.15.



Gambar 4.2.15 Diagram Aktivitas Memberi Penilaian dan Ulasan Produk Pesanan

xv. Melihat Detail Pemesanan

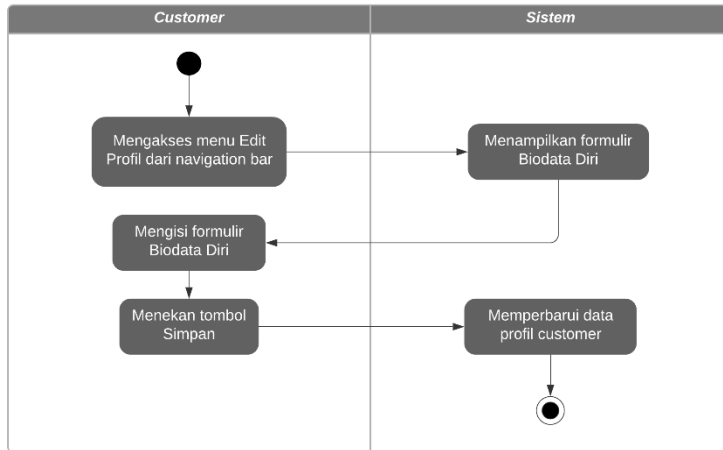
Detail pemesanan mencakup rekapitulasi produk yang dibeli, detail pengiriman seperti nomor resi untuk pelacakan pengiriman, dan detail pembayaran. Tahapan fitur melihat detail pemesanan dapat dilihat pada Gambar 4.2.16.



Gambar 4.2.16 Diagram Aktivitas Melihat Detail Pemesanan

xvi.     **Mengelola Profil**

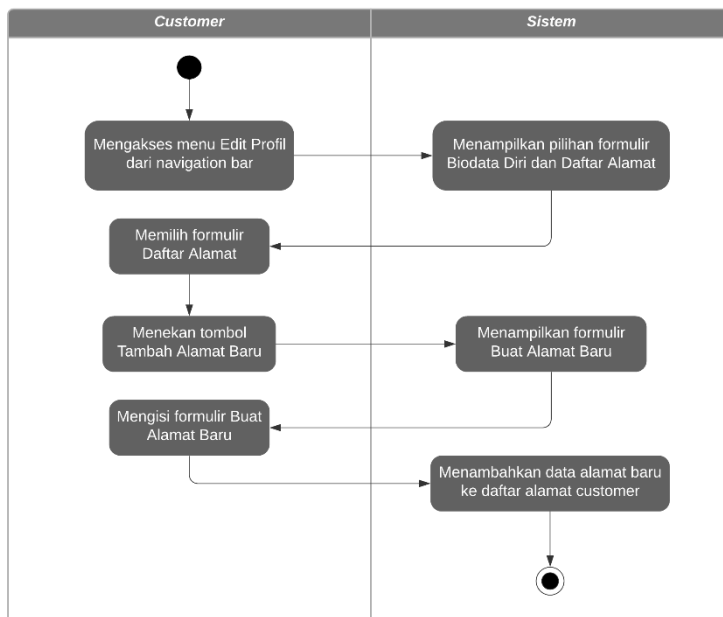
Manajemen profil bagi *customer* dilakukan untuk memperbarui biodata seperti nama, nomor telepon dan alamat yang bisa digunakan sebagai alamat pengiriman. Tahapan fitur mengelola profil dapat dilihat pada Gambar 4.2.17.



*Gambar 4.2.17 Diagram Aktivitas Mengelola Profil*

xvii. Menambah Alamat Pengiriman

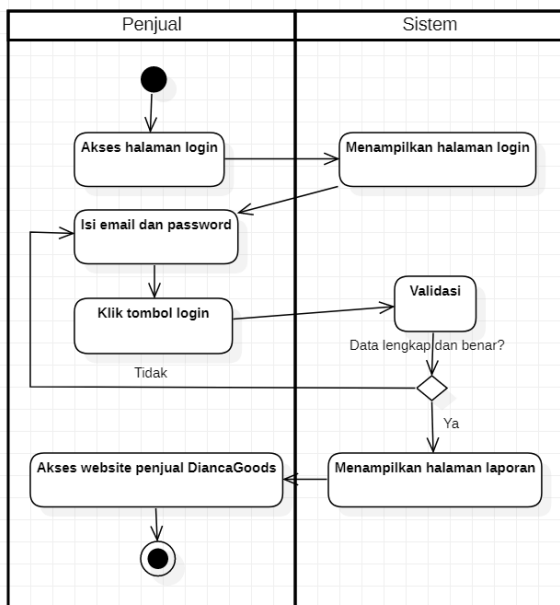
*Customer* dapat memiliki lebih dari satu alamat pengiriman. Alamat dapat ditambahkan di manajemen profil, atau saat melakukan *checkout* di bagian Tambah Alamat Baru. Tahapan fitur menambah alamat pengiriman dapat dilihat pada Gambar 4.2.18.



Gambar 4.2.18 Diagram Aktivitas Menambah Alamat Pengiriman

xviii. Login Penjual

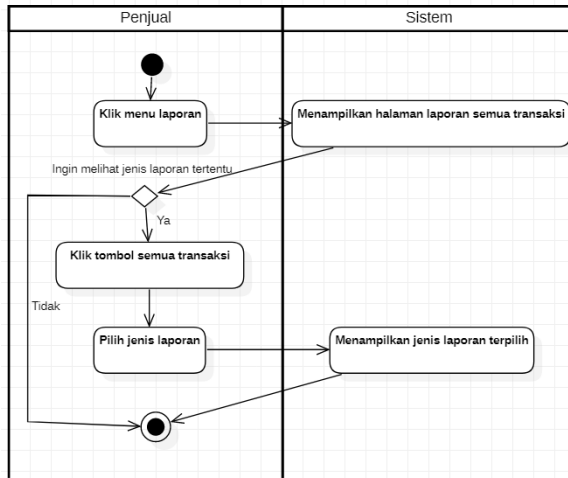
Toko melalui administratornya memiliki akun yang dapat digunakan untuk melakukan rekapitulasi jual beli yang berlangsung di aplikasi. Administrator ini dapat melakukan login ke sistem untuk mengakses *endpoint* administrator. Tahapan fitur login penjual dapat dilihat pada Gambar 4.2.19.



Gambar 4.2.19 Diagram Aktivitas Login Penjual

xix. Melihat Laporan

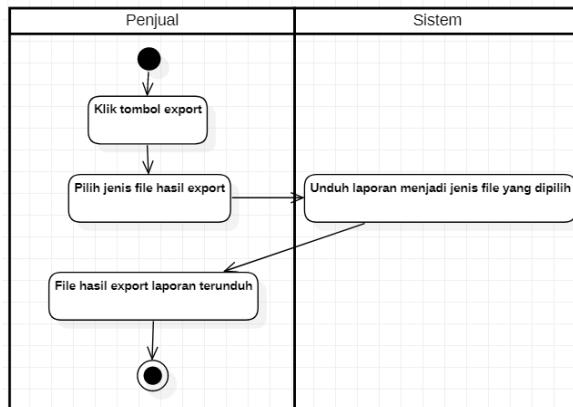
Administrator dapat melihat laporan melalui menu laporan. Laporan yang dapat dihasilkan antara lain laporan transaksi, laporan total produk terjual, dan laporan pemasukan. Tahapan fitur melihat laporan dapat dilihat pada Gambar 4.2.20.



Gambar 4.2.20 Diagram Aktivitas Melihat Laporan

xx. Ekspor File Laporan

Administrator dapat mengeksport laporan yang ditampilkan dalam bentuk dashboard ke dalam bentuk PDF atau XLSX. Tahapan fitur ekspor file laporan dapat dilihat pada Gambar 4.2.21.

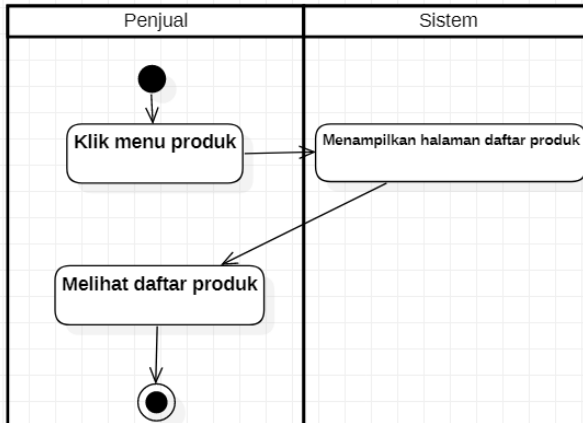


Gambar 4.2.21 Diagram Aktivitas Ekspor File Laporan



xxi. Melihat Daftar Produk

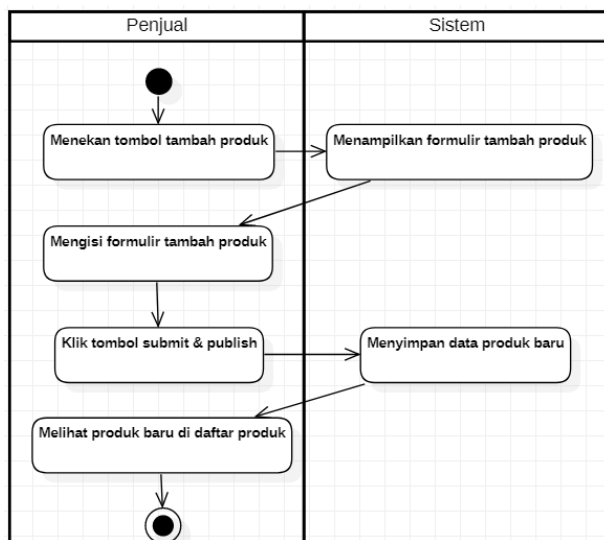
Produk-produk ditampilkan dalam bentuk tabel dengan menu di tiap barisnya untuk mengubah detail produk, serta tombol untuk menambah produk baru. Tahapan fitur melihat daftar produk dapat dilihat pada Gambar 4.2.22.



Gambar 4.2.22 Diagram Aktivitas Melihat Daftar Produk

## xxii. Menambah Produk

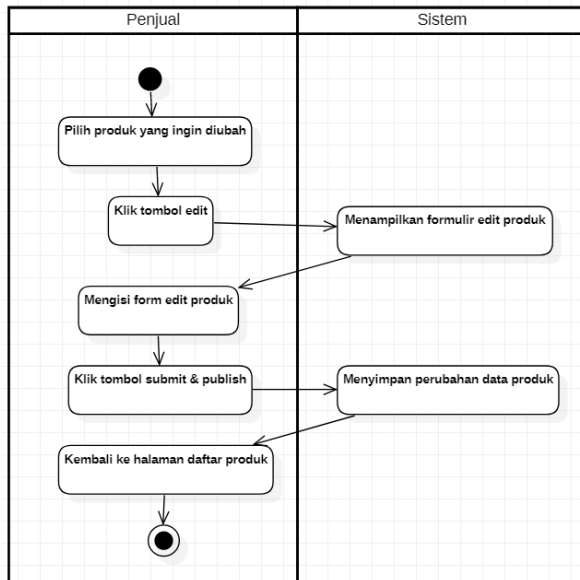
Penambahan produk baru dilakukan dengan mengisi formulir keterangan produk. Dalam menambah produk baru, administrator secara otomatis menambah varian produk baru. Hal ini dikarenakan yang akan diproses dalam *checkout* adalah varian produk, yang memiliki keterangan harga satuan dan berat, bukan produknya. Tahapan fitur menambah produk dapat dilihat pada Gambar 4.2.23.



Gambar 4.2.23 Diagram Aktivitas Menambah Produk

xxiii. Mengubah Data Produk

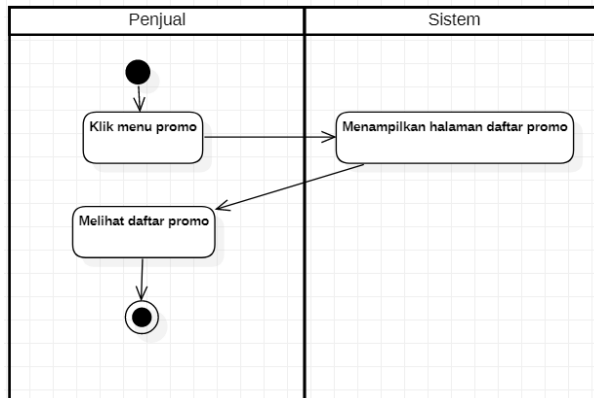
Keterangan produk dapat diubah dengan form yang sama seperti form penambahan produk baru. Tahapan fitur mengubah data produk dapat dilihat pada Gambar 4.2.24.



Gambar 4.2.24 Diagram Aktivitas Mengubah Data Produk

xxiv. Melihat Daftar Promo

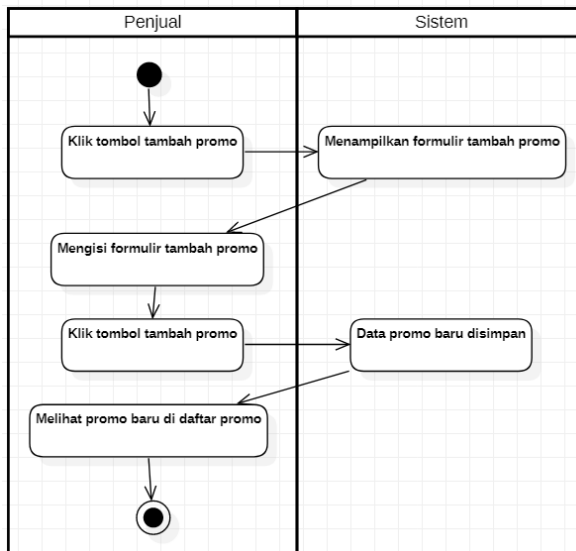
Administrator dapat melihat daftar promo yang telah dibuat di sistem, baik berstatus aktif (di-*publish*) maupun tidak. Tahapan fitur melihat daftar promo dapat dilihat pada Gambar 4.2.25.



Gambar 4.2.25 Diagram Aktivitas Melihat Daftar Promo

xxv. Menambah Promo

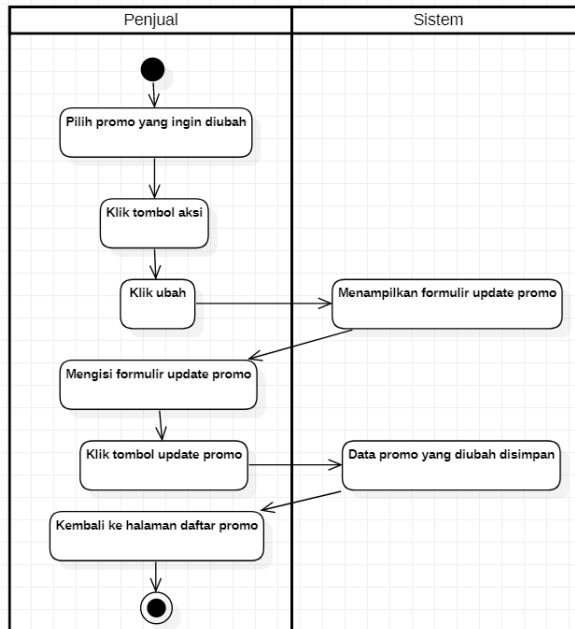
Penambahan promo dilakukan dengan mengisi formulir penambahan promo yang berisi nama promo, jenis, promo berupa persentase atau nominal, waktu mulai dan berakhir promo, serta status aktifnya. Tahapan fitur menambah promo dapat dilihat pada Gambar 4.2.26.



Gambar 4.2.26 Diagram Aktivitas Menambah Promo

xxvi. Mengubah Promo

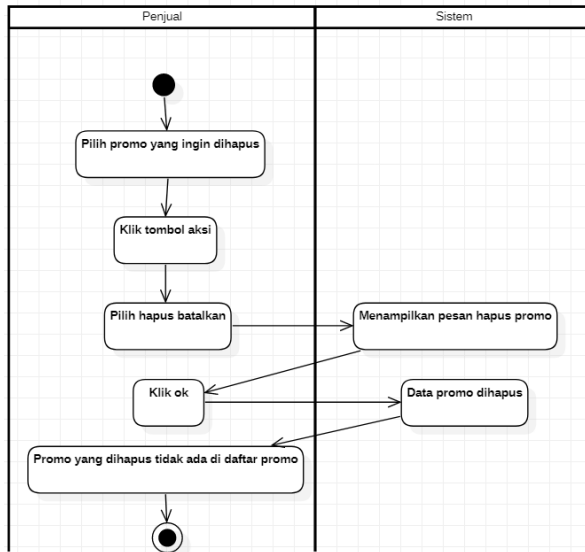
Perubahan keterangan promo dapat dilakukan dengan mengisi formulir yang sama seperti penambahan promo. Tahapan fitur mengubah promo dapat dilihat pada Gambar 4.2.27.



Gambar 4.2.27 Diagram Aktivitas Mengubah Promo

xxvii. Menghapus Promo

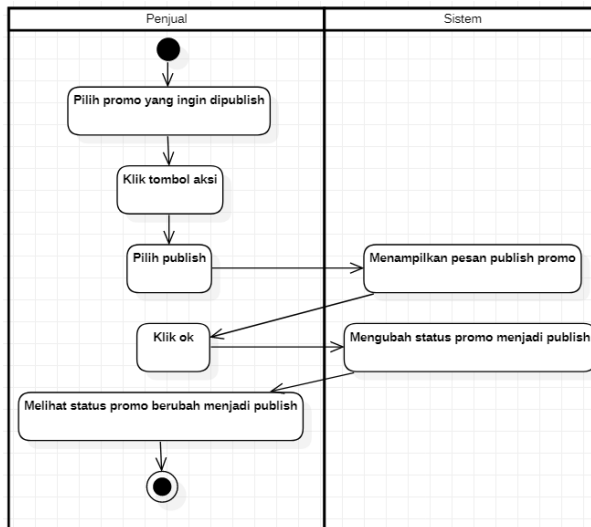
Promo yang tidak akan digunakan lagi dapat dihapus dari daftar promo oleh administrator. Tahapan fitur menghapus promo dapat dilihat pada Gambar 4.2.28.



Gambar 4.2.28 Diagram Aktivitas Menghapus Promo

xxviii.      *Mem-publish Promo*

Apabila status promo adalah *published*, maka promo akan secara otomatis ditampilkan di *endpoint* customer dalam rentang waktu dari waktu mulai hingga waktu berakhirnya promo tersebut. Tahapan fitur *mem-publish* promo dapat dilihat pada Gambar 4.2.29.

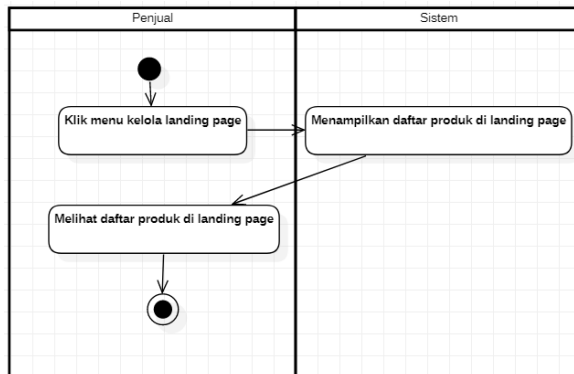


Gambar 4.2.29 Diagram Aktivitas *Mem-publish Promo*



xxix. Melihat Daftar Produk di Landing Page

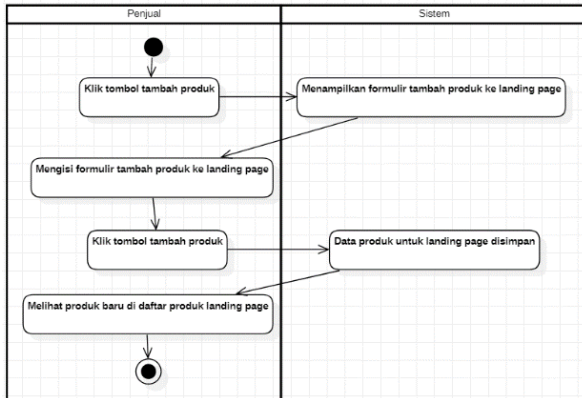
Administrator dapat melihat produk-produk yang tampil di halaman awal, seperti produk *best sellers* dan produk fitur. Tahapan fitur melihat daftar produk di landing page dapat dilihat pada Gambar 4.2.30.



Gambar 4.2.30 Diagram Aktivitas Melihat Daftar Produk di Landing Page

xxx. Menambah Produk di Landing Page

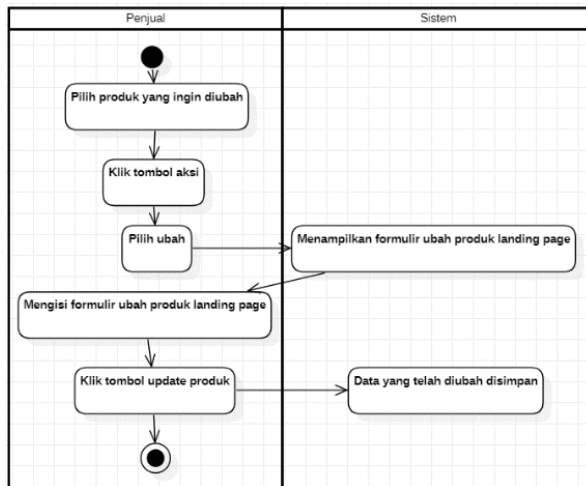
Administrator dapat menambah produk *best sellers* atau produk fitur yang ditampilkan di halaman awal dengan melakukan pemilihan produk yang diinginkan. Tahapan fitur menambah produk di landing page dapat dilihat pada Gambar 4.2.31.



Gambar 4.2.31 Diagram Aktivitas Menambah Produk di Landing Page

xxxi.      Mengubah Produk di Landing Page

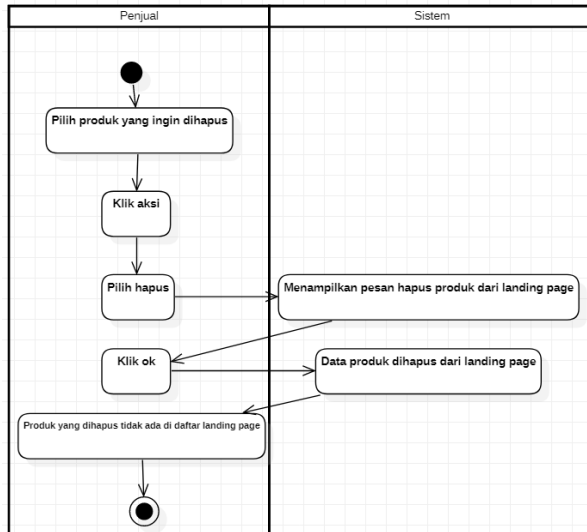
Perubahan produk yang ditampilkan di halaman awal dimungkinkan dengan mengisi formulir perubahan. Tahapan fitur mengubah produk di landing page dapat dilihat pada Gambar 4.2.32.



Gambar 4.2.32 Diagram Aktivitas Mengubah Produk di Landing Page

xxxii. Menghapus Produk di Landing Page

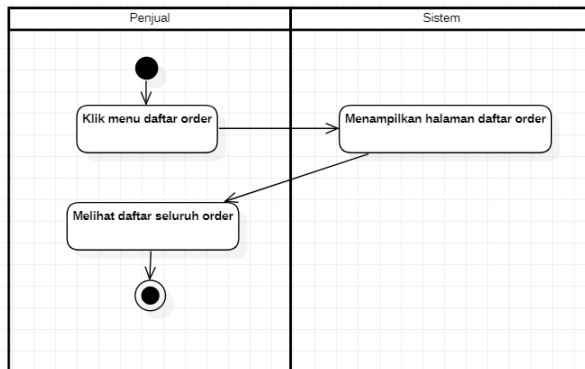
Apabila produk ingin dihapus dari halaman awal, administrator cukup menekan tombol hapus. Tahapan fitur menghapus produk di landing page dapat dilihat pada Gambar 4.2.33.



Gambar 4.2.33 Diagram Aktivitas Menghapus Produk di Landing Page

xxxiii. Melihat Daftar Order

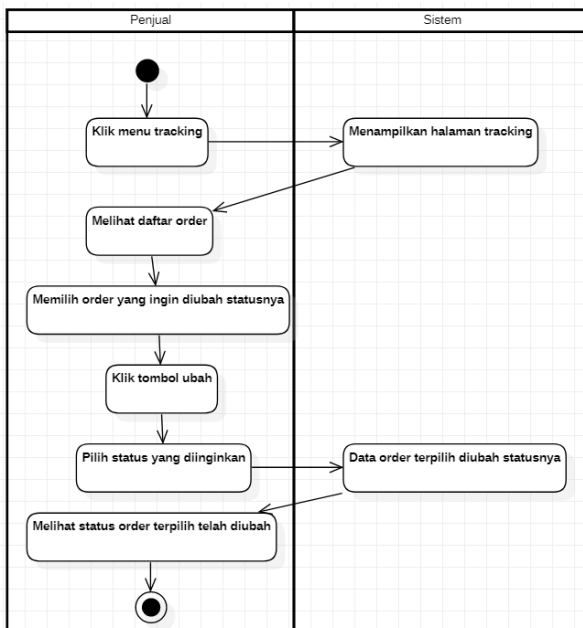
Segala pesanan yang terbuat di *endpoint customer* ditampilkan di *endpoint administrator* dalam bentuk tabel. Tiap pesanan memiliki aksi ubah status yang dapat dilakukan administrator langsung dari tabel daftar pesanan. Tahapan fitur melihat daftar order dapat dilihat pada Gambar 4.2.34.



Gambar 4.2.34 Diagram Aktivitas Melihat Daftar Order

xxxiv. Mengubah Status Order

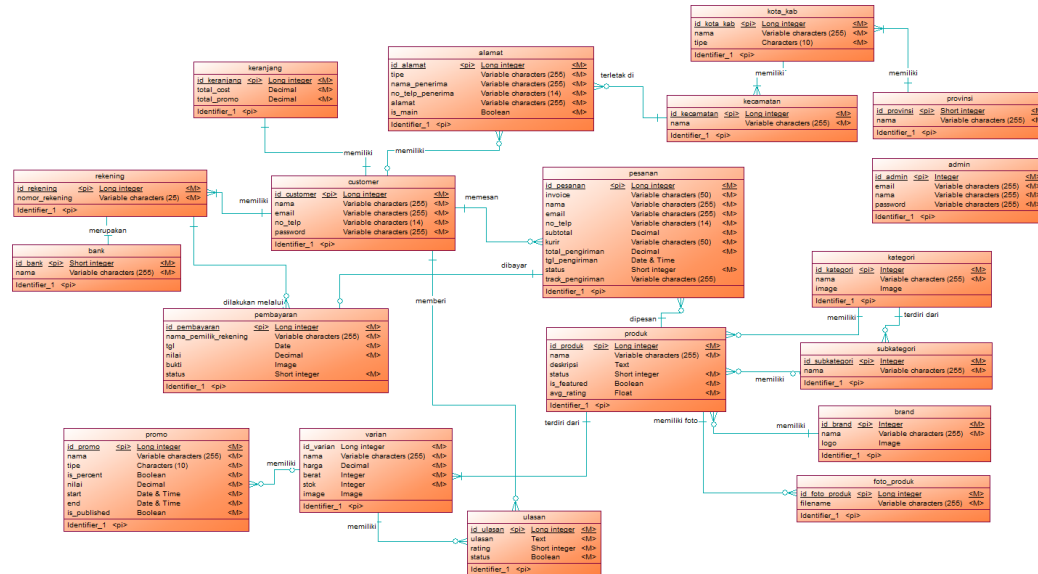
Perubahan status pesanan dilakukan dengan memilih status yang tepat dari pilihan status yang ada. Tahapan fitur mengubah status order dapat dilihat pada Gambar 4.2.35.



Gambar 4.2.35 Diagram Aktivitas Mengubah Status Order

### 4.2.3 Conceptual Data Model

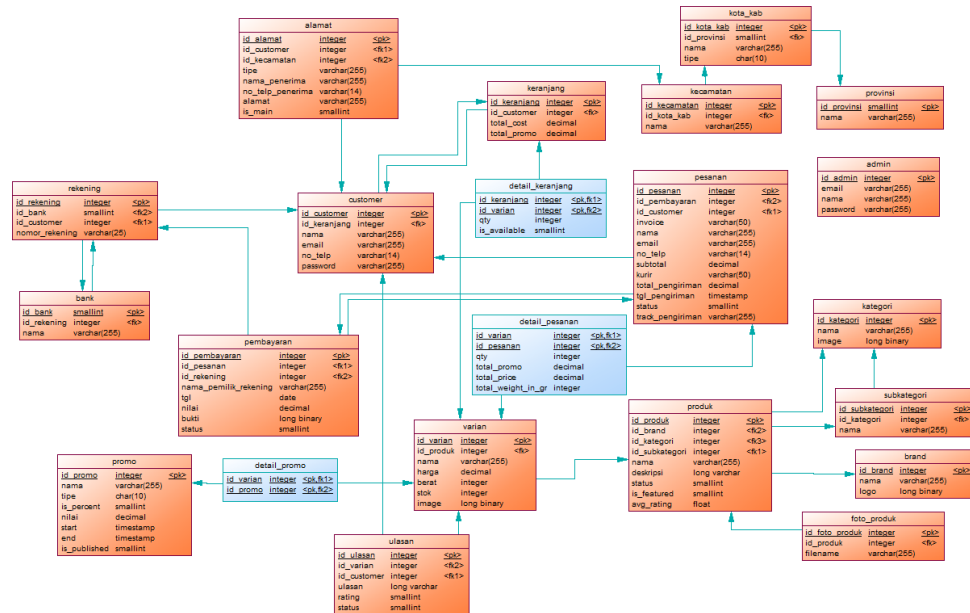
Conceptual Data Model dari sistem toko online ini digambarkan pada Gambar 4.2.36.



*Gambar 4.2.36 Conceptual Data Model*

## 4.2.4 Physical Data Model

Physical Data Model dari sistem toko online ini digambarkan pada Gambar 4.2.37.



Gambar 4.2.37 Physical Data Model



## BAB V

### IMPLEMENTASI SISTEM

Bab ini membahas tentang implementasi dari sistem yang kami buat. Penjelasan implementasi terbagi menjadi dua bagian, yaitu implementasi controller dan implementasi antarmuka yang dihasilkan. Kedua jenis implementasi akan terbagi lagi menjadi dua berdasarkan endpoint pengguna, yaitu customer dan administrator.

#### 5.1 Implementasi Controller

*Controller* berfungsi untuk menampung pengolahan logika yang digunakan di aplikasi dan bagaimana data dapat berubah untuk setiap proses yang dilakukan di aplikasi.

##### 5.1.1 Endpoint Customer

Di *endpoint customer*, modul controller terbagi menjadi sembilan, di antaranya Autentikasi, Landing Page, Pencarian dan Kategori, Detail Produk, Keranjang, Checkout dan Pembayaran, Profil, Penilaian dan Ulasan, dan Daftar Transaksi.

###### 5.1.1.1 Autentikasi

```
public function login(Request $request)
{
    $this->validate($request, [
        'email' => 'required|email|exists:customers,email',
        'password' => 'required|string'
    ]);

    $customer = Customer::where('email', $request->email)->first();

    if(!$customer) {
        return redirect()->back()->with(['error' => 'Email salah.']);
    }

    $credentials = $request->only('email', 'password');

    if(Auth::guard('customer')->attempt($credentials)) {
        return redirect()->intended(route('home'));
    } else {
        return redirect()->back()->with(['error' => 'Password salah.']);
    }
}

public function register(Request $request){
    $this->validate($request, [
```

```

        'name' => 'required|string|max:100',
        'email' => 'required|email',
        'password' => 'required|confirmed',
        'phone' => 'required|numeric',
    ));

    $cust_email = Customer::where('email', $request->email)->first();

    if ($cust_email != NULL) {
        return redirect()->back()->with(['error' => 'Akun sudah terdaftar, silakan Login']);
    }

    $customer = Customer::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => $request->password,
        'phone_number' => $request->phone,
        'activate_token' => Str::random(4)
    ]);

    Cart::create([
        'customer_id' => $customer->id,
        'total_cost' => 0
    ]);

    $credentials = $request->only('email', 'password');

    if ($customer->status == false) {
        return redirect()->route('customer.sms_verification', ['id' => $customer->id]);
    } else if (Auth::guard('customer')->attempt($credentials)) {
        return redirect()->intended(route('home'));
    }
}

public function showVerificationForm($id)
{
    $customer = Customer::where('id', $id)->first();

    $customer->update(['activate_token' => Str::random(4)]);

    Nexmo::message()->send([
        'to' => $customer->phone,
        'from' => 'Vonage APIs',
        'text' => 'Verification code : ' . $customer->activate_token
    ]);

    return view('verify', compact('customer', 'id'));
}

public function verify(Request $request)
{
    $this->validate($request, [
        'digit_a' => 'required',
        'digit_b' => 'required',
        'digit_c' => 'required',
        'digit_d' => 'required',
        'customer_id' => 'required|exists:customers,id'
    ]);

    $customer = Customer::where('id', $request->customer_id)->first();

    $a = str_split($customer->activate_token, 1);

    if ($request->digit_a == $a[0] && $request->digit_b == $a[1] &&
        $request->digit_c == $a[2] && $request->digit_d == $a[3]) {
        $customer->update(['status' => true]);

        $cart = Cart::create([
            'customer_id' => $customer->id,
            'total_cost' => 0
        ]);

        Auth::login($customer);
    }
}

```

```

        return redirect()->intended(route('home'));
    }
    return redirect()->back()->with(['error' => 'Kode verifikasi salah.']);
}

```

### 5.1.1.2 Landing Page

```

public function index()
{
    $newproducts = Product::where('status', 1)
        ->where('is_featured', 1)->with('variant', 'images')
        ->orderBy('created_at', 'ASC')->limit(5)->get();
    $bestseller = Product::where('status', 1)
        ->where('is_featured', 2)->with('variant', 'images')
        ->orderBy('created_at', 'ASC')->limit(3)->get();
    $featured = Product::where('status', 1)
        ->where('is_featured', 3)->with('variant', 'images')
        ->orderBy('created_at', 'ASC')->limit(3)->get();

    $category = Category::with('subcategory')
        ->orderBy('created_at', 'ASC')->limit(4)->get();

    $promos = Promo::with('details.variant.product.images')
        ->where('is_published', 1)
        ->where(DB::raw("CONCAT(`end_date`, ' ', `end_time`)", '>=', Carbon::now()-
        >toDateTimeString())
        ->where(DB::raw("CONCAT(`start_date`, ' ', `start_time`)", '<=', Carbon::now()-
        >toDateTimeString())
        ->paginate(4);

    $str = NULL;

    $reviews = Review::with('product', 'customer')
        ->orderBy('created_at', 'ASC')->limit(10)->get();

    return view('home', compact('category', 'newproducts', 'bestseller', 'featured', 'str',
    'promos', 'reviews'));
}

```

### 5.1.1.3 Keranjang

```

public function addToCart(Request $request)
{
    if(Auth::guard('customer')->check()){
        $this->validate($request, [
            'product_variant_id' => 'required|int',
            'qty' => 'required|int'
        ]);

        $cart = Cart::with('details.variant.product.images')->where('customer_id',
        Auth::guard('customer')->user()->id)->first();

        $variant = ProductVariant::where('id', $request->product_variant_id)->first();

        if($variant->stock > 0) {
            if(!CartDetail::where('cart_id', $cart->id)->where('product_variant_id', $variant-
            >id)->exists()) {
                $cart_detail = CartDetail::create([
                    'cart_id' => $cart->id,
                    'product_variant_id' => $variant->id,
                    'qty' => $request->qty,
                    'price' => ($variant->price) * $request->qty,
                    'promo' => $variant->promo_price
                ]);

                $cart->total_cost += $cart_detail->price;
                $cart->save();
            }
        }
    }
}

```

```

    } else {
        $cart_variant = CartDetail::where('cart_id', $cart->id)-
>where('product_variant_id', $variant->id)->first();
        $cart_variant->qty += $request->qty;
        $cart_variant->price += ($variant->price) * $request->qty;
        $cart_variant->promo += $variant->promo_price;
        $cart_variant->save();

        $cart->total_cost = CartDetail::where('cart_id', $cart->id)->sum('price');
        $cart->total_promo = CartDetail::where('cart_id', $cart->id)->sum('promo');
        $cart->save();
    }

    return redirect()->back()->with(['success' => 'Produk berhasil ditambahkan ke
keranjang.']);
}

    return redirect()->back()->with(['error' => 'Produk tidak dapat ditambahkan ke
keranjang.']);
}
    return redirect(route('customer.login'));
}

```

### 5.1.1.4 Checkout dan Pembayaran

```

public function checkout(Request $request)
{
    if(Auth::guard('customer')->check()) {
        if (!$request->session()->has('id_item_cart')) {
            return redirect()->route('home');
        }

        $cart = Cart::where('customer_id', auth()->guard('customer')->user()->id)->first();
        $cart_arr = $request->session()->get('id_item_cart');
        $total_cost = 0;
        $promos = 0;
        $cart_detail = array();
        $str = NULL;

        foreach($cart_arr as $cd) {
            $cart_detail[] = CartDetail::with('variant.product')->where('is_avail', 1)-
>where('id', $cd)->first();
        }

        foreach($cart_detail as $cd) {
            $total_cost += $cd->price;
            $promos += $cd->promo;
        }

        $address = Address::with('district.city.province')->where('customer_id', auth()-
>guard('customer')->user()->id);
        if ($request->session()->has('id_address')) {
            $address = $address->where('id', $request->session()->get('id_address'));
        } else {
            $address = $address->where('is_main', 1);
        }
        $address = $address->first();
        $provinces = Province::get();

        return view('checkout', compact('cart', 'cart_detail', 'address', 'total_cost',
'promos', 'provinces', 'str'));
    }
}

public function address(Request $request)
{
    if(Auth::guard('customer')->check()) {
        $this->validate($request, [
            'address_type' => 'required|string',
            'receiver_name' => 'required|string',
            'receiver_phone' => 'required|string',

```

```

        'district_id' => 'required|exists:districts,id',
        'postal_code' => 'required|string',
        'address' => 'required|string',
        'is_main' => 'boolean',
        'customer_id' => 'required|exists:customers,id'
    ));

    $countAddress = Address::where('customer_id', auth()->guard('customer')->user()->id)-
>count();

    $is_main = 0;

    if ($request->is_main == 1) {
        Address::where('customer_id', auth()->guard('customer')->user()->id)-
>update(['is_main'=>0]);
        $is_main = 1;
    }
    if ($countAddress == 0) {
        $is_main = 1;
    }

    $address = Address::create([
        'province_id' => $request->province_id,
        'city_id' => $request->city_id,
        'address_type' => $request->address_type,
        'receiver_name' => $request->receiver_name,
        'receiver_phone' => $request->receiver_phone,
        'district_id' => $request->district_id,
        'postal_code' => $request->postal_code,
        'address' => $request->address,
        'is_main' => $is_main,
        'customer_id' => $request->customer_id
    ]);

    $customer = Customer::where('id', auth()->guard('customer')->user()->id)->first();
    $customer->update([
        'address' => 1
    ]);

    $customer->save();

    return json_encode($address);
}
}

public function payment(Request $request)
{
    if(Auth::guard('customer')->check()) {
        $this->validate($request, [
            'courier' => 'required',
            'duration' => 'required',
            'address_id' => 'required|exists:addresses,id',
            'shipping_cost' => 'required',
            'subtotal' => 'required',
            'promo' => 'integer'
        ]);

        $subtotal = $request->subtotal;
        $shipping_cost = $request->shipping_cost;
        $address_id = $request->address_id;
        $total_cost = $request->subtotal + $request->shipping_cost;
        $courier = $request->courier;
        $duration = $request->duration;
        $promos = $request->promo;
        $str = NULL;
        return view('payment', compact('subtotal', 'shipping_cost', 'address_id',
'total_cost', 'courier', 'duration', 'promos', 'str'));
    }
}

public function paymentDone($id)
{
    if(Auth::guard('customer')->check()) {

```

```

        $order = Order::with('payment')->where('customer_id', Auth::guard('customer')->user()->id)->where('id', $id)->first();
        $str = NULL;
        return view('payment-done', compact('order', 'str'));
    }
}

```

### 5.1.1.5 Profil

```

public function index()
{
    if(auth()->guard('customer')->check()) {
        $customer = Customer::with('address')->where('id', auth()->guard('customer')->user()->id)->first();
        $str = NULL;
        return view('profile', compact('customer', 'str'));
    }
}

public function editProfile(Request $request)
{
    if(auth()->guard('customer')->check()) {
        $validator = Validator::make($request->all(), [
            'name' => 'required',
            'birthday' => 'required',
            'gender' => 'required',
            'phone_number' => 'required'
        ]);

        $customer = Customer::find(auth()->guard('customer')->user()->id);

        $customer->update([
            'name' => $request->name,
            'birthday' => $request->birthday,
            'gender' => $request->gender,
            'phone_number' => $request->phone_number
        ]);

        return redirect(route('profile'));
    }
}

```

### 5.1.1.6 Daftar Transaksi

```

public function index($status, Request $request)
{
    if(Auth::guard('customer')->check()) {
        if($status != 5) {
            $orders = Order::with('details.variant.product.images', 'payment')->where('customer_id', Auth::guard('customer')->user()->id)->where('status', $status)->orderBy('created_at', 'DESC')->get();
        } else {
            $orders = Order::with('details.variant.product.images', 'payment')->where('customer_id', Auth::guard('customer')->user()->id)->orderBy('created_at', 'DESC')->get();
        }
        $str = NULL;
        return view('transaction-list', compact('orders', 'str'));
    }
    return redirect(route('customer.login'));
}

```

## 5.1.2 Endpoint Admin Penjual

Di endpoint administrator, modul *controller* terbagi menjadi empat, di antaranya Laporan, Manajemen Landing Page, Manajemen Order, dan Manajemen Promo.

### 5.1.2.1 Laporan

```
public function allReport()
{
    if(Auth::guard('web')->check()) {
        $start = Carbon::now()->startOfMonth()->format('Y-m-d H:i:s');
        $end = Carbon::now()->endOfMonth()->format('Y-m-d H:i:s');

        if (request()->from_date != '' && request()->to_date != '') {
            $start = Carbon::parse(request()->from_date)->format('Y-m-d') . ' 00:00:01';
            $end = Carbon::parse(request()->to_date)->format('Y-m-d') . ' 23:59:59';
        }

        $total_order = Order::whereBetween('created_at', [$start, $end])->count();

        $monthly_income = Order::where('status', 1)->orWhere('status', 2)->orWhere('status', 3)->orWhere('status', 4)->orWhere('status', 5)->whereBetween('created_at', [$start, $end])->sum('subtotal');

        $finished = Order::whereBetween('updated_at', [$start, $end])->get();
        $monthly_sold = 0;
        foreach ($finished as $f) {
            $monthly_sold += OrderDetail::where('order_id', $f->id)->sum('qty');
        }

        $sold = ProductVariant::with('product')->whereBetween('created_at', [$start, $end])->whereRaw('stock <= 0')->count();

        $orders = Order::with('payment')->whereBetween('created_at', [$start, $end])->orderBy('created_at', 'ASC')->get();
        return view('report.all-report', compact('orders', 'monthly_income', 'monthly_sold', 'total_order', 'sold', 'start', 'end'));
    }
    return redirect(route('administrator.login'));
}

public function allReportPDF()
{
    if(Auth::guard('web')->check()) {
        $start = Carbon::parse(request()->from_date)->format('Y-m-d') . ' 00:00:01';
        $end = Carbon::parse(request()->to_date)->format('Y-m-d') . ' 23:59:59';

        $orders = Order::with('payment')->whereBetween('created_at', [$start, $end])->orderBy('created_at', 'ASC')->get();

        $total_order = Order::whereBetween('created_at', [$start, $end])->count();

        $monthly_income = Order::whereBetween('created_at', [$start, $end])->sum('subtotal');

        view()->share('orders', $orders);
        $pdf = PDF::loadView('report.all-report-print', ['orders'=>$orders, 'start'=>$start, 'end'=>$end, 'total_order'=>$total_order, 'monthly_income'=>$monthly_income]);
        return $pdf->download('laporan-transaksi.pdf');
    }
    return redirect(route('administrator.login'));
}

public function allReportExcel()
{
    if(Auth::guard('web')->check()) {
        $start = Carbon::parse(request()->from_date)->format('Y-m-d') . ' 00:00:01';
```

```

$end = Carbon::parse(request()->to_date)->format('Y-m-d') . ' 23:59:59';

return Excel::download(new TransactionReport($start, $end), 'laporan-transaksi.xlsx');
}
return redirect(route('administrator.login'));
}

```

### 5.1.2.2 Manajemen Produk

```

public function addProduct(Request $request)
{
    if(Auth::guard('web')->check()) {
        try {
            $this->validate($request, [
                'name' => 'required|string',
                'category_id' => 'required|integer|exists:categories,id',
                'subcategory_id' => 'integer|exists:subcategories,id',
                'brand_id' => 'required|integer|exists:brands,id',
                'rate' => 'required',
                'description' => 'string',
                'image_product' => 'required',
                'image_product.*' => 'image|mimes:jpg,jpeg,png',
                'variants' => 'array',
                'variants.*' => 'integer|exists:product_variants,id'
            ]);

            $product = Product::create([
                'category_id' => $request->category_id,
                'subcategory_id' => $request->subcategory_id,
                'brand_id' => $request->brand_id,
                'name' => $request->name,
                'slug' => Str::slug($request->name),
                'description' => $request->description,
                'rate' => $request->rate
            ]);

            if ($request->hasFile('image_product')) {
                foreach ($request->file('image_product') as $image) {
                    $name = time() . Str::slug($request->name) . '-' . $product->id . '.' .
$image->getClientOriginalExtension();
                    $image->storeAs('public/products', $name);

                    $product_image = ProductImage::create([
                        'product_id' => $product->id,
                        'filename' => $name
                    ]);
                }
            }

            foreach($request->variants as $v) {
                $variant = ProductVariant::where('id', $v)->first();
                $variant->product_id = $product->id;
                $variant->save();
            }

            return redirect(route('administrator.products'))->with(['success' => 'Produk dan
Varian Baru Ditambahkan!']);
        } catch(\Exception $e) {
            return redirect()->back()->with(['error' => $e->getMessage()]);
        }
    }
    return redirect(route('administrator.login'));
}

public function storeVariant(Request $request)
{
    if (Auth::guard('web')->check()) {
        $this->validate($request, [
            'name' => 'required|string',
            'price' => 'required|integer',
            'weight' => 'required|integer',

```



```

        'stock' => 'required|integer',
        'image_variant' => 'image|mimes:jpg,jpeg,png'
    ));

    $product_variant = ProductVariant::create([
        'name' => $request->name,
        'price' => $request->price,
        'weight' => $request->weight,
        'stock' => $request->stock,
    ]);

    if ($request->hasFile('image_variant')) {
        $file = $request->file('image_variant');
        $name = time() . $product_variant->id . Str::slug($request->name) . '.' . $file-
>getClientOriginalExtension();
        $file->storeAs('public/variants', $name);

        $product_image = ProductImage::create([
            'product_variant_id' => $product_variant->id,
            'filename' => $name
        ]);

        $product_variant->image = $product_image->filename;
        $product_variant->save();
    }

    return json_encode($product_variant);
}
}

public function storeCategory(Request $request)
{
    if (Auth::guard('web')->check()) {
        try {
            $this->validate($request, [
                'name' => 'required|string',
                'image_category' => 'required|image|mimes:png,jpeg,jpg'
            ]);

            if ($request->hasFile('image_category')) {
                $image = $request->file('image_category');
                $name = time() . Str::slug($request->name) . '.' . $image-
>getClientOriginalExtension();
                $image->storeAs('public/categories', $name);

                $category = Category::create([
                    'name' => $request->name,
                    'slug' => $request->name,
                    'image' => $name
                ]);
            }
            return redirect()->back()->with(['success' => "Kategori Ditambahkan!"]);
        } catch (\Exception $e) {
            return redirect()->back()->with(['error' => $e->getMessage()]);
        }
    }
}
}

```

### 5.1.2.3 Manajemen Order

```

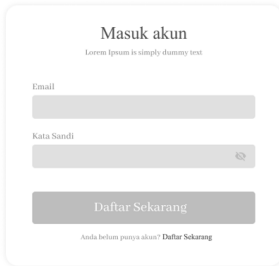
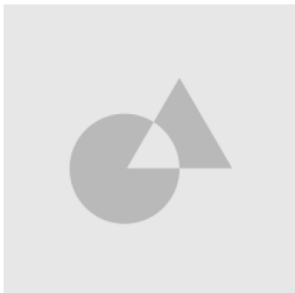
public function showOrders()
{
    if (Auth::guard('web')->check()) {
        $orders = Order::with('address')->orderBy('created_at', 'DESC')->paginate(7);
        return view('admin.orders', compact('orders'));
    }
    return redirect(route('administrator.login'));
}

```

## 5.2 Implementasi Antarmuka

### 5.2.1 Endpoint Customer

#### 5.2.1.1 Autentikasi



Masuk akun  
Lorem Ipsum is simply dummy text

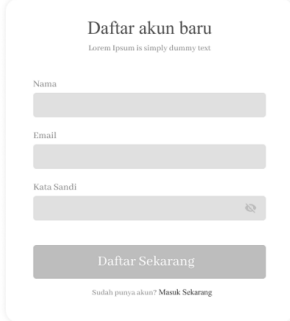
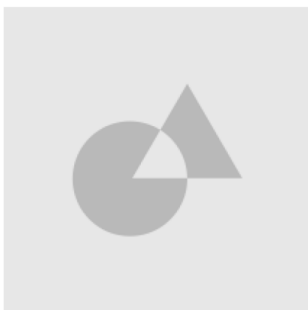
Email

Kata Sandi

Daftar Sekarang

Anda belum punya akun? [Daftar Sekarang](#)

*Gambar 5.2.1 Halaman Login*



Daftar akun baru  
Lorem Ipsum is simply dummy text

Nama

Email

Kata Sandi

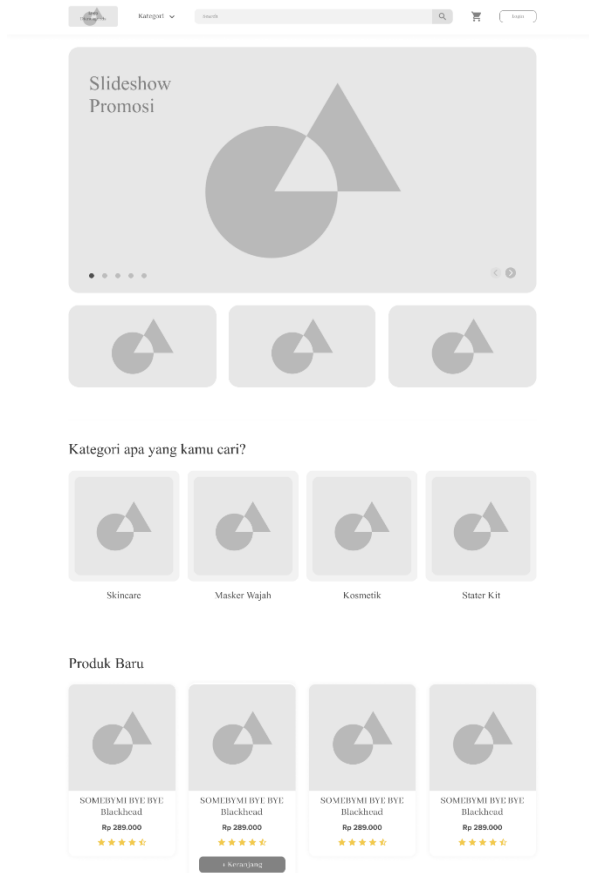
Daftar Sekarang

Sudah punya akun? [Masuk Sekarang](#)

*Gambar 5.2.2 Halaman Registrasi*

Gambar 5.2.1 dan Gambar 5.2.2 berisi tampilan halaman autentikasi *customer*.

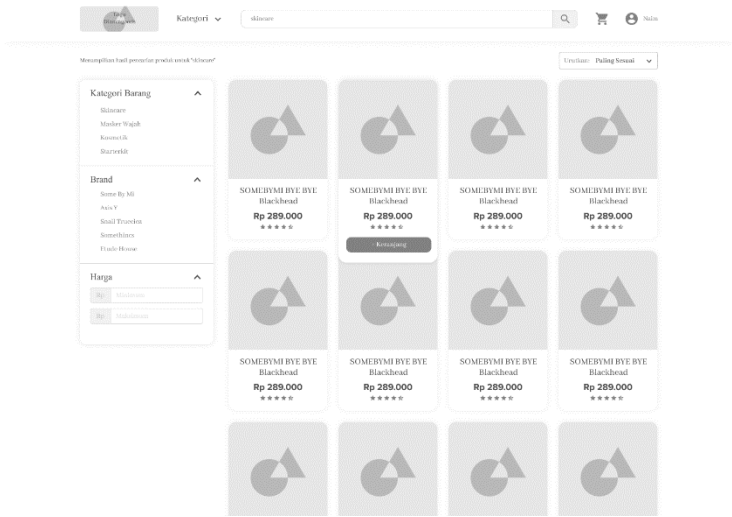
### 5.2.1.2 Landing Page



*Gambar 5.2.3 Halaman Awal*

Gambar 5.2.3 berisi tampilan halaman awal atau landing page yang akan terlihat oleh *customer*, baik yang sudah terautentikasi maupun yang belum.

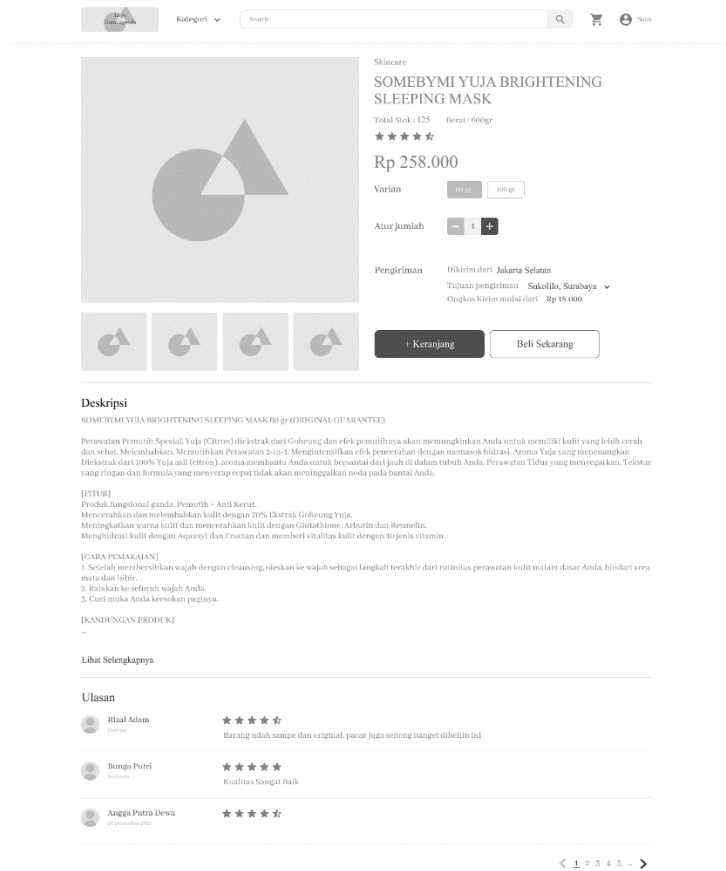
5.2.1.3 Pencarian dan Kategori



Gambar 5.2.4 Halaman Pencarian Kategori

Gambar 5.2.4 menampilkan halaman hasil pencarian berdasarkan kategori. Di bagian samping terdapat filter tambahan seperti brand yang dapat dipilih oleh *customer*.

5.2.1.4 Detail Produk

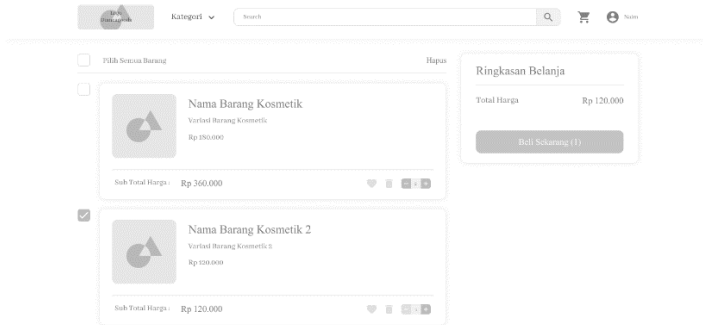


Gambar 5.2.5 Halaman Detail Produk

Gambar 5.2.5 merupakan tampilan halaman detail produk yang menampilkan nama produk, varian produk, harga tiap varian, kuantitas yang diinginkan, dan tujuan pengiriman untuk mengalkulasi biaya pengiriman. *Customer* dapat menambahkan

produk ke keranjang dengan menekan tombol ‘+ Keranjang’ atau langsung *checkout* dengan menekan tombol ‘Beli Sekarang’.


#### 5.2.1.5 Keranjang



*Gambar 5.2.6 Halaman Keranjang*

Gambar 5.2.6 menampilkan halaman keranjang di mana *customer* dapat memilih produk mana saja yang akan dilanjutkan ke *checkout* pemesanan.


5.2.1.6 Checkout dan Pembayaran




Kategori

Search

Q

 0

 Nama

Pembayaran


1 dari 2 langkah

1 Checkout

2 Bayar

Alamat Pengiriman

Buat Alamat Baru



Nama Barang Kosmetik 2

Varian (Barang Kosmetik 2)

1 Barang (1 kg)

Sub Total Harga

Rp 120.000

Go

Ringkasan Belanja

Total Harga

Rp 120.000

Ongkos Kirim

Rp 17.000

Total Tagihan

Rp 137.000

Pilih Pembayaran

Pengiriman

Pilih Jasa Pengiriman

JPNT

▼

Pilih Durasi

Regular (4-5 hari)

Ringkasan Pengiriman

Jasa Pengiriman

(JPNT (Reguler))


Domeski

4 - 5 Hari

Estimasi Tiba

21 - 23 Nov 2020


Gambar 5.2.7 Halaman Checkout 1: Keterangan Pengiriman




Kategori

Search

Q

 0

 Nama

Pembayaran

1 dari 2 langkah

1 Checkout

2 Bayar

Pilih Metode Pembayaran

Transfer Bank

Pilih Bank

Bank Negara Indonesia

▼

Ringkasan Belanja

Total Harga

Rp 120.000

Ongkos Kirim

Rp 17.000

Total Tagihan

Rp 137.000

Bayar (1)

Gambar 5.2.8 Halaman Checkout 2: Pembayaran

87

**Selesaikan Pembayaran**

Nomor Pemesanan  
DG01232020AV

Metode Pembayaran  
Transfer Bank BNI

Nomor Rekening  
800 152 6846

Atas Nama  
Toko Diancagoods

Batas Pembayaran  
Senin, 29 Desember 2020, Pukul 16:21 WIB

Total Pembayaran  
Rp 137.002

Transfer tepat sesuai 3 digit terakhir agar uangmu tetap proses validasi

Upload Bukti Pembayaran

Petunjuk Pembayaran

ATM BNI

BNI Mobile

*Gambar 5.2.9 Halaman Checkout 3: Selesaikan Pembayaran*

Gambar 5.2.7, Gambar 5.2.8 dan Gambar 5.2.9 menampilkan halaman sekuens *checkout* yang harus dilewati *customer* saat ingin menempatkan pesanan.

### 5.2.1.7 Profil

**Profil**

Biodata Diri

Daftar Alamat

Rekening Bank

**Ubah Biodata Diri**

Nama

Tanggal Lahir

Jenis Kelamin

**Ubah Kontak**

Email

Nomor HP

Jaga akun anda agar tetap aman dengan Verifikasi nomor ponsel

Untuk keamanan transaksi anda, silahkan tambahkan nomor yang dapat kami hubungi.

*Gambar 5.2.10 Halaman Manajemen Profil*



Gambar 5.2.10 merupakan tampilan halaman profil *customer* di mana terdapat data diri yang dapat diubah dan alamat yang dapat ditambah.

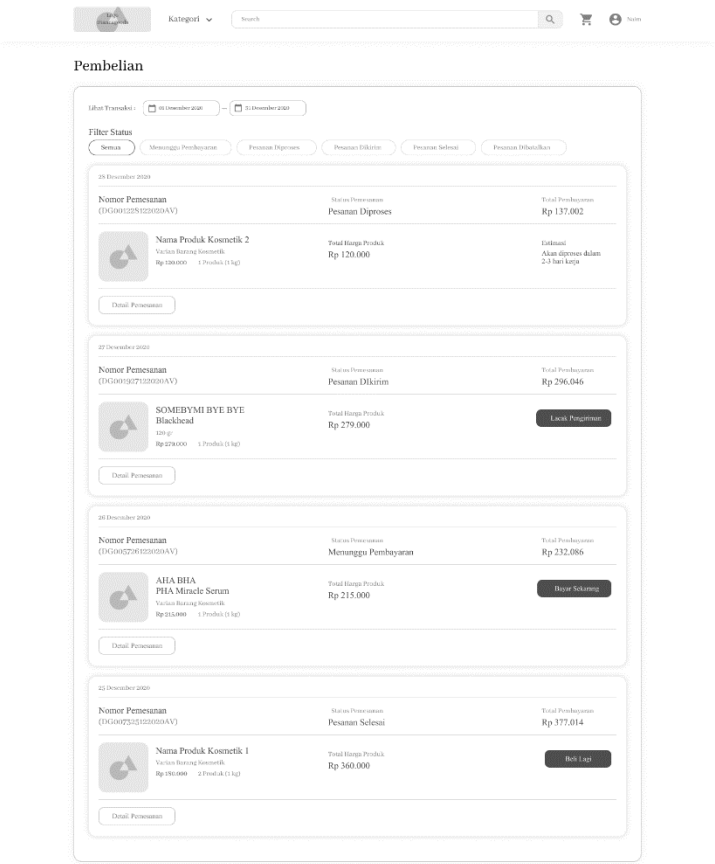
### 5.2.1.8 Penilaian dan Ulasan

The screenshot displays a web interface for product reviews. At the top, there is a navigation bar with a logo, a 'Kategori' dropdown menu, a search bar, and icons for a shopping cart and user profile. Below the navigation bar, a red button labeled 'Ulasan Saya' is visible. The main content area shows two identical review forms for the product 'AHA BHA PHA Miracle Serum'. Each form includes the order number 'DG01232020AV' and the purchase date 'Senin, 12 April 2021 17.00 WIB'. The product name is followed by a small image, a link to 'Variasi Berlangganan', and a link to 'Pengalaman Kualitas Layanan?'. Below this, there is a star rating system with five stars, where the first four are filled and the fifth is empty. A text input field for the review is provided, with a placeholder text 'Berikan ulasan tentang...'. At the bottom of each form are two buttons: 'Batal' and 'Kirim'.

Gambar 5.2.11 Halaman Penilaian dan Ulasan

Gambar 5.2.11 menampilkan halaman penilaian dan ulasan di mana *customer* dapat memberikan penilaiannya terhadap produk yang telah dipesan.

5.2.1.9 Daftar Transaksi



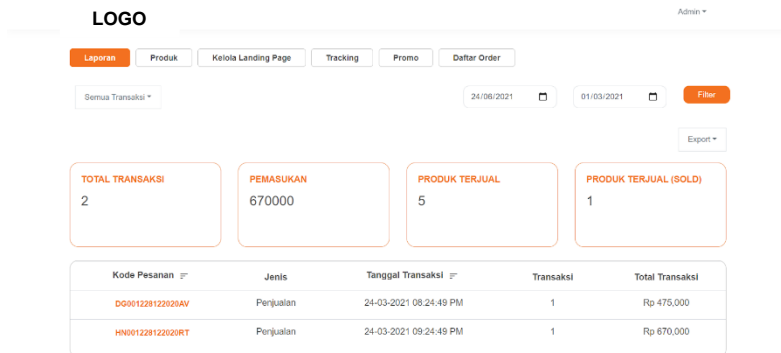
Gambar 5.2.12 Halaman Daftar Transaksi

Gambar 5.2.12 merupakan tampilan halaman daftar transaksi atau pemesanan yang dapat diakses *customer*. Halaman ini dapat diatur berdasarkan status pesanan. Komponen *card* untuk tiap pesanan berbeda berdasarkan statusnya. Sebagai contoh,

apabila masih berupa ‘Menunggu Pembayaran’, maka tombol yang terlihat adalah ‘Selesaikan Pembayaran’ dan apabila status pesanan adalah ‘Selesai’, maka tombol yang terlihat dan berfungsi adalah ‘Beli Lagi’ dan ‘Beri Ulasan’.

## 5.2.2 Endpoint Admin Penjual

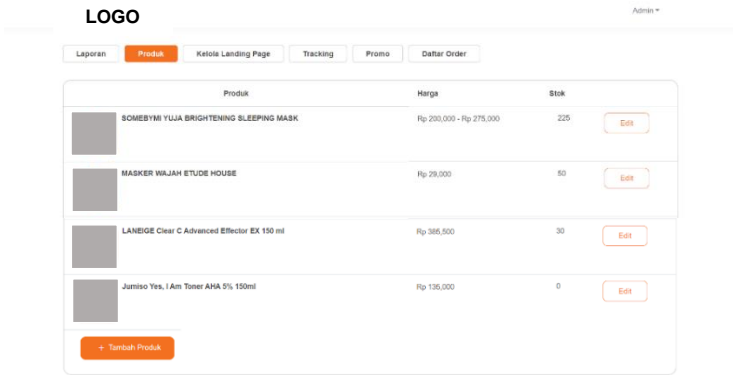
### 5.2.2.1 Laporan



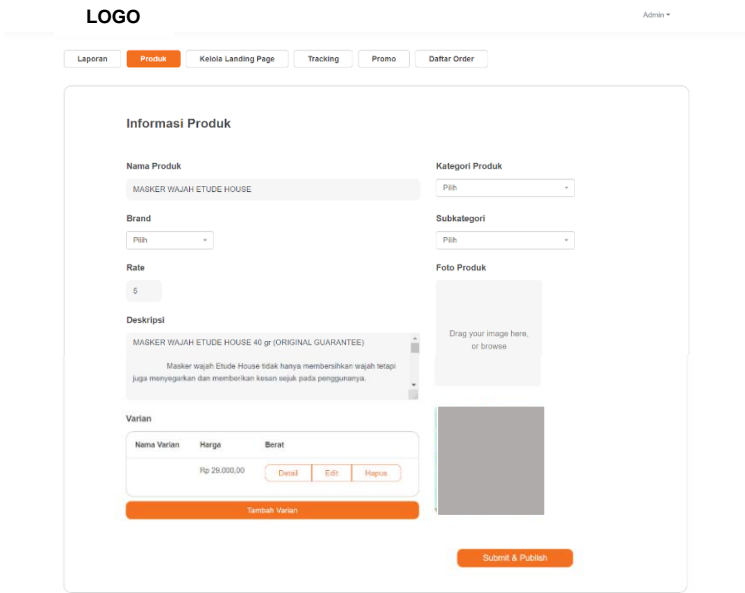
Gambar 5.2.13 Halaman Overview Laporan Admin

Gambar 5.2.13 merupakan tampilan dashboard admin yang berisi informasi performa penjualan toko selama rentang waktu tertentu yang dapat diatur di bagian atas. Menu untuk mengekspor laporan menjadi PDF atau XLSX terdapat sebagai *dropdown*.

5.2.2.2 Manajemen Produk



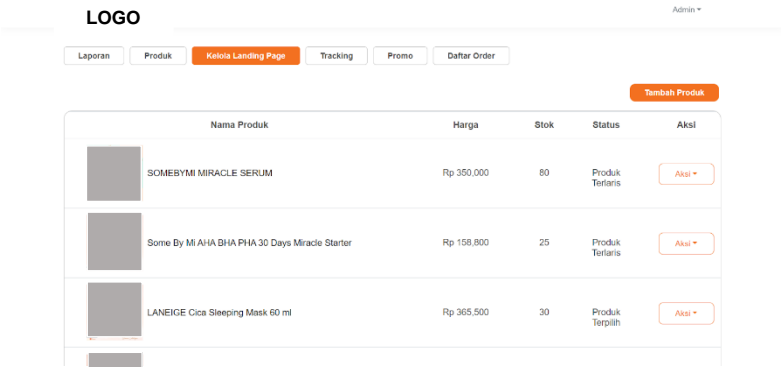
Gambar 5.2.14 Halaman Manajemen Produk: Daftar Produk



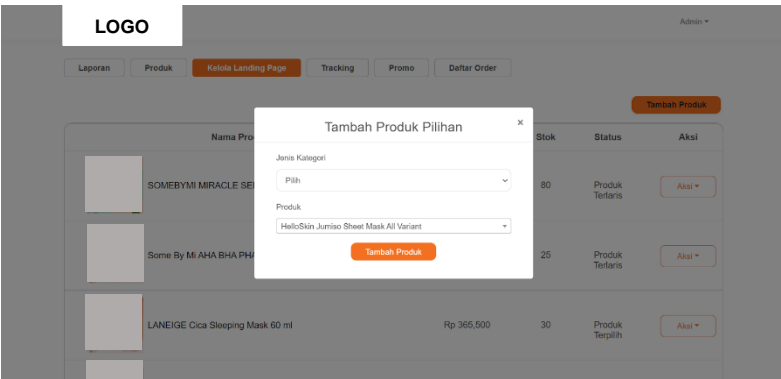
Gambar 5.2.15 Halaman Manajemen Produk: Edit Produk

Gambar 5.2.15 dan Gambar 5.2.15 menampilkan halaman manajemen produk yang dapat diakses oleh administrator. Di halaman daftar produk, terdapat pilihan untuk menambah produk baru.

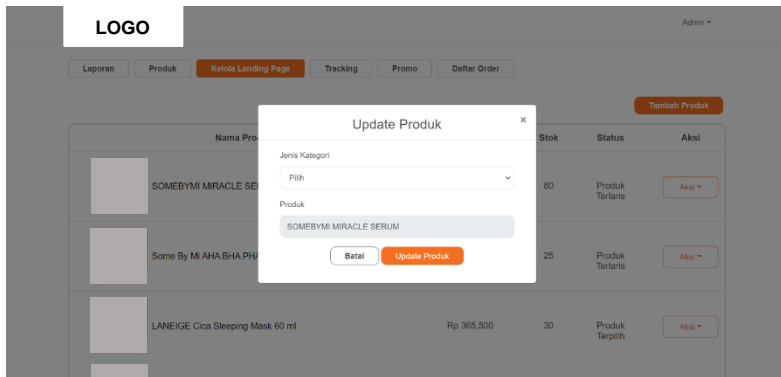
5.2.2.3 Manajemen Landing Page



Gambar 5.2.16 Halaman Manajemen Landing Page: Daftar Produk yang Ditampilkan di Landing Page



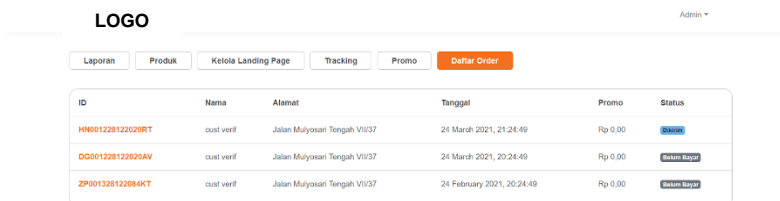
Gambar 5.2.17 Halaman Manajemen Landing Page: Tambah Produk untuk Ditampilkan di Landing Page



Gambar 5.2.18 Halaman Manajemen Landing Page: Update Produk di Bagian Landing Page

Gambar 5.2.16 menampilkan halaman pengaturan *landing page* atau halaman awal. Seperti Gambar 5.2.17 dan Gambar 5.2.18, tiap produk dapat diatur untuk ditampilkan di halaman awal aplikasi sebagai produk fitur atau produk *best seller*.

## 5.2.2.4 Manajemen Order



Gambar 5.2.19 Halaman Manajemen Order: Daftar Order

LOGO

Admin ▾

Laporan

Produk

Kelola Landing Page

Tracking

Promo

Daftar Order

Semua 4

Belum Bayar 1

Perlu Dibayar 1

Dibayar 1

Selesai 1

Dibatalkan 1

ID	Nama	Alamat	Tanggal	Status	
HN001228122020RT	cust verif		24 March 2021, 21:24:49	Selesai	Ubah ▾
DG001228122020AV	cust verif		24 March 2021, 20:24:49	Belum Bayar	Ubah ▾
ZP001328122020KT	cust verif		24 February 2021, 20:24:49	Belum Bayar	Ubah ▾

Gambar 5.2.20 Halaman Manajemen Order: Keterangan Status Tracking Order

Gambar 5.2.20 dan Gambar 5.2.20 menampilkan *overview* pemesanan yang dilakukan oleh *customer* dan aksi-aksi yang dapat dilakukan oleh administrator, yaitu mengubah status pemesanan melalui *dropdown* ‘Ubah’, mengecek bukti pembayaran dengan mengklik nomor ID pesanan yang akan membuka halaman detail pesanan, dan memasukkan nomor resi atau URL pelacakan pengiriman pesanan di halaman detail pesanan.

### 5.2.2.5 Manajemen Promo

LOGO

Admin ▾

Laporan

Produk

Kelola Landing Page

Tracking

Promo

Daftar Order

Semua Promo

Flash Sale

Event Promo

Promo Item

No	Waktu Mulai	Waktu Selesai	Jumlah Produk	Status	
1	2021-03-11 - 00:00:00 WIB	2021-03-13 - 00:00:00 WIB	3	Belum Publish	Aksi ▾
2	2021-03-11 - 00:00:00 WIB	2021-03-13 - 00:00:00 WIB	0	Belum Publish	Aksi ▾
3	2021-03-11 - 00:00:00 WIB	2021-03-13 - 00:00:00 WIB	0	Belum Publish	Aksi ▾
					+ Tambah Promo

Gambar 5.2.21 Halaman Manajemen Promo: Daftar Promo

**LOGO**

Laporan Produk Kelola Lan

Semua Promo Flash Sale Even

No	Waktu Mulai
1	2021-03-11 - 00:00:00 WIB
2	2021-03-11 - 00:00:00 WIB
3	2021-03-11 - 00:00:00 WIB

**Tambah Promo**

Jenis Promo  
Pilih

Nama Promo

Batas Waktu Pembayaran Setelah Checkout  
menit

Tanggal Mulai Jam Mulai  
2021-06-24 00:00:00

Tanggal Selesai Jam Selesai  
2021-06-24 00:00:00

Jenis Diskon  
Persentase

Promo

Admin

➕ Tambah Promo

*Gambar 5.2.22 Halaman Manajemen Promo: Tambah Promo*

Gambar 5.2.22 dan Gambar 5.2.22 menampilkan halaman manajemen promo oleh administrator.



## **BAB VI**

### **PENGUJIAN DAN EVALUASI**

#### **6.1 Tujuan Pengujian**

Pengujian dilakukan terhadap aplikasi Toko Online guna menguji kesesuaian dan ketepatan fungsionalitas dari seluruh sistem aplikasi.

#### **6.2 Kriteria Pengujian**

Penilaian atas pencapaian tujuan pengujian didapatkan dengan memerhatikan beberapa hasil yang diharapkan berikut ini:

- i. Kemampuan aplikasi menampilkan fitur untuk pembeli mulai dari melakukan login, melihat produk, melakukan checkout, memberikan penilaian dan ulasan, serta mengelola profil.
- ii. Kemampuan aplikasi untuk menampilkan fitur untuk penjual mulai dari melakukan login, melihat laporan, mengelola produk, mengelola landing page, mengelola promo, dan mengelola order.
- iii. Kesesuaian dalam memenuhi kebutuhan non-fungsional aplikasi yaitu:
  - a. Pengaksesan sistem dari area yang terhubung internet.
  - b. Sistem memiliki antarmuka yang mudah dipahami.
  - c. Sistem dapat diakses dari browser manapun.
  - d. Data dalam sistem terlindungi dari pihak yang tidak berwenang.

#### **6.3 Skenario Pengujian**

Skenario pengujian dilakukan dengan melakukan peran sebagai pembeli dan penjual yang akan menjalankan fitur-fitur dan

seluruh kebutuhan fungsional dari sistem. Langkah-langkah untuk setiap kebutuhan fungsional yaitu sebagai berikut:

#### **6.3.1 Pembeli**

- a. Pembeli dapat membuat akun baru
- b. Pembeli dapat melakukan login
- c. Pembeli dapat melihat daftar produk di kategori
- d. Pembeli dapat mencari produk berdasarkan nama
- e. Pembeli dapat melihat detail produk
- f. Pembeli dapat menambahkan produk ke keranjang
- g. Pembeli dapat melihat keranjang
- h. Pembeli dapat mengubah kuantitas produk di keranjang
- i. Pembeli dapat melakukan checkout
- j. Pembeli dapat menentukan alamat pengiriman
- k. Pembeli dapat menentukan metode pembayaran
- l. Pembeli dapat melihat riwayat pemesanan
- m. Pembeli dapat melakukan konfirmasi pembayaran
- n. Pembeli dapat memberi penilaian dan ulasan produk pesanan
- o. Pembeli dapat mengelola profil

#### **6.3.2 Penjual**

- a. Penjual dapat melakukan login ke website penjual
- b. Penjual dapat melihat laporan
- c. Penjual dapat melakukan export laporan sebagai PDF atau Excel
- d. Penjual dapat melihat daftar produk
- e. Penjual dapat menambah produk
- f. Penjual dapat mengubah produk

- g. Penjual dapat melihat daftar promo
- h. Penjual dapat menambah promo
- i. Penjual dapat mengubah promo
- j. Penjual dapat menghapus promo
- k. Penjual dapat melakukan publish promo
- l. Penjual dapat melihat daftar produk di landing page
- m. Penjual dapat menambah produk di landing page
- n. Penjual dapat mengubah produk di landing page
- o. Penjual dapat menghapus produk di landing page
- p. Penjual dapat melihat daftar order
- q. Penjual dapat mengubah status order

## 6.4 Evaluasi Pengujian

Pada Tabel 6.4.1, terdapat hasil pengujian kebutuhan fungsional system yang telah dilakukan:

Skenario Pengujian	Hasil
Pembeli dapat membuat akun baru	Berhasil
Pembeli dapat melakukan <i>login</i>	Berhasil
Pembeli dapat melihat daftar produk di kategori	Berhasil
Pembeli dapat mencari produk berdasarkan nama	Berhasil
Pembeli dapat melihat detail produk	Berhasil
Pembeli dapat menambahkan produk ke keranjang	Berhasil
Pembeli dapat melihat keranjang	Berhasil

Pembeli dapat mengubah kuantitas produk di keranjang	Berhasil
Pembeli dapat melakukan <i>checkout</i>	Berhasil
Pembeli dapat menentukan alamat pengiriman	Berhasil
Pembeli dapat menentukan metode pembayaran	Berhasil
Pembeli dapat melihat riwayat pemesanan	Berhasil
Pembeli dapat melakukan konfirmasi pembayaran	Berhasil
Pembeli dapat memberi penilaian dan ulasan produk pesanan	Berhasil
Pembeli dapat mengelola profil	Berhasil
Penjual dapat melakukan <i>login</i> ke <i>website</i> penjual	Berhasil
Penjual dapat melihat laporan	Berhasil
Penjual dapat melakukan <i>export</i> laporan sebagai PDF atau Excel	Berhasil
Penjual dapat melihat daftar produk	Berhasil
Penjual dapat menambah produk	Berhasil
Penjual dapat mengubah produk	Berhasil
Penjual dapat melihat daftar promo	Berhasil
Penjual dapat menambah promo	Berhasil
Penjual dapat mengubah promo	Berhasil

Penjual dapat menghapus promo	Berhasil
Penjual dapat melakukan <i>publish</i> promo	Berhasil
Penjual dapat melihat daftar produk di <i>landing page</i>	Berhasil
Penjual dapat menambah produk di <i>landing page</i>	Berhasil
Penjual dapat mengubah produk di <i>landing page</i>	Berhasil
Penjual dapat menghapus produk di <i>landing page</i>	Berhasil
Penjual dapat melihat daftar <i>order</i>	Berhasil
Penjual dapat mengubah status <i>order</i>	Berhasil

*Tabel 6.4.1 Evaluasi Pengujian*

*[Halaman ini sengaja dikosongkan]*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1 Kesimpulan**

Kesimpulan yang didapat setelah melakukan pengembangan aplikasi pada kegiatan Kerja Praktik di Departemen Teknik Informatika adalah sebagai berikut:

- i. Aplikasi yang dibangun berguna untuk membantu proses jual beli Toko Produk Kecantikan.
- ii. Dengan adanya aplikasi ini, pembeli dapat melakukan pembelian produk dari Toko Produk Kecantikan secara daring.
- iii. Dengan adanya aplikasi ini, penjual dapat melihat laporan dan memantau transaksi jual-beli secara *real time*.
- iv. Berdasarkan hasil pengujian fungsionalitas yang telah dilakukan, fitur-fitur pada aplikasi ini telah berhasil dibuat dan dijalankan sesuai dengan kebutuhan fungsional utama dari sistem.

#### **7.2 Saran**

Aplikasi memungkinkan untuk dikembangkan menjadi lebih *mobile-first*, dimulai dari penerapan antarmuka yang lebih ramah bagi *mobile device*. Selain itu, untuk meningkatkan kualitas pengalaman pengguna, aplikasi bisa diintegrasikan ke sistem pembayaran virtual dan pelacakan pemesanan *in-app*.

*[Halaman ini sengaja dikosongkan]*



## DAFTAR PUSTAKA

- [1] “HTML - Wikipedia,” [Online]. Available:  
<https://en.wikipedia.org/wiki/HTML>.
- [2] R. Nixon, Learning PHP, MySQL & JavaScript: With  
jQuery, CSS & HTML5, O'Reilly Media, Inc, 2014.
- [3] D. Flanagan, JavaScript: The Definitive Guide: Master the  
World's Most-Used Programming Language, O'Reilly Media,  
2020.
- [4] M. Stauffer, Laravel: Up & Running: A Framework for  
Building Modern PHP Apps, O'Reilly Media, Inc., 2019.
- [5] “MySQL - Wikipedia,” [Online]. Available:  
<https://en.wikipedia.org/wiki/MySQL>.
- [6] M. Fowler dan J. Highsmith, “The agile manifesto,” dalam  
*Software Development*, 2001, pp. 28-32.

*[Halaman ini sengaja dikosongkan]*

## **BIODATA PENULIS I**

Nama : Rosa Valentine Lammora  
Tempat, Tanggal Lahir : Jakarta, 14 Februari 2001  
Jenis Kelamin : Perempuan  
Email : [rosalammora@gmail.com](mailto:rosalammora@gmail.com)

### **PENDIDIKAN FORMAL**

2018 – sekarang : S1 Teknik Informatika ITS  
2015 – 2018 : SMAN 1 Kota Bekasi

### **KEMAMPUAN**

- Web Programming (HTML, CSS, Javascript, Laravel, CodeIgniter)
- Programming (C, C++, Python)
- Database Management (MySQL, PostgreSQL, SQL Server)
- Project Management

### **AKADEMIS**

Program Studi : Departemen Informatika –  
Fakultas Teknologi Elektro dan  
Informatika Cerdas  
Angkatan : 2018  
Semester : 6 (enam)

*[Halaman ini sengaja dikosongkan]*

## **BIODATA PENULIS II**

Nama : Afia Hana Yusriya  
Tempat, Tanggal Lahir : Surabaya, 3 April 2000  
Jenis Kelamin : Perempuan  
Email : [afiaahanaa@gmail.com](mailto:afiaahanaa@gmail.com)

### **PENDIDIKAN FORMAL**

2018 – sekarang : S1 Teknik Informatika ITS  
2015 – 2018 : SMAN 5 Kota Surabaya

### **KEMAMPUAN**

- Web Programming (HTML, CSS, Javascript, Laravel)
- Programming (C, C++, Python)
- Database Management (MySQL, PostgreSQL, SQL Server)
- Project Management

### **AKADEMIS**

Program Studi : Departemen Informatika –  
Fakultas Teknologi Elektro dan  
Informatika Cerdas  
Angkatan : 2018  
Semester : 6 (enam)

*[Halaman ini sengaja dikosongkan]*